DOCUMENT RESUME

ED 372 752 IR 016 798

AUTHOR	Huston, Rick, Ed.; Armel, Donald, Ed.
TITLE	Association of Small Computer Users in Education (ASCUE) Summer Conference. Proceedings (27th, North Myrtle Beach, South Carolina, June 12-16, 1994).
INSTITUTION	Association of Small Computer Users in Education.
PUB DATE	Jun 94
NOTE	349p.; For 1992 proceedings, see ED 357 732.
PUB TYPE	Collected Works - Conference Proceedings (021)
EDRS PRICE	MF01/PC14 Plus Postage.
DESCRIPTORS	Computer Assisted Instruction; Computer Networks; Computer Simulation; *Computer Uses in Education; Educational Technology; Ethics; Expert Systems; Futures (of Society); Higher Education; Information Networks; Microcomputers; Professional Associations; Technological Advancement; *Telecommunications
IDENTIFIERS	Internet

ABSTRACT

Topics addressed by 40 papers from a conference on microcomputers include: developing a campus wide computer ethics policy; integrating new technologies into professional education; campus computer networks; computer assisted instruction; client/server architecture; competencies for entry-level computing positions; auditing and professional development; mobile computing; computer applications in physics; artificial neural networks; virtual classrooms; the North Carolina Information Highway; Computer Science curriculum; expert systems; ethics for information systems professionals; listservs; local area networks; computer security; computer simulations; telecommunications and education; campus-wide information systems (CWIS); and the Internet. This volume also includes information about ASCUE; a list of ASCUE board members; and a presenters index. Most of the papers include references. (JLB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 372 752

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it
☐ Minor changes have been made to improve reproduction quality
● Points of view or opinions stated in this document do not necessarily represent official OERI position or policy

ASCUE

ASSOCIATION OF SMALL COMPUTER USERS IN EDUCATION

"Our Second Quarter Century of Resourse Sharing"

Proceedings of the 1994 ASCUE Summer Conference

27th Annual Conference
June 12-16, 1994

North Myrtle Beach, South Carolina

Edited by:
[illegible] University of South Carolina - Aiken
[illegible] Western Illinois University

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
C. P. Singer
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

BEST COPY AVAILABLE

ASCUE

ASSOCIATION OF SMALL COMPUTER USERS IN EDUCATION

"Our Second Quarter Century of Resourse Sharing"

Proceedings of the 1994 ASCUE Summer Conference

27th Annual Conference
June 12-16, 1994

North Myrtle Beach, South Carolina

Edited by:
Rick Huston, University of South Carolina - Aiken
Donald Armel, Eastern Illinois University

Association of Small Computer Users in Education

"Our Second Quarter Century of Resource Sharing"

Proceedings of the 1994 ASCUE Summer Conference
27th Annual Conference
June 12-16, 1994
Myrtle Beach, South Carolina

ABOUT ASCUE

ASCUE, the Association of Small Computer Users in Education, is a group of people interested in small college computing issues. It is a blend of people from all over the country who use computers in their teaching, academic support, and administrative support functions. Begun in 1968 as CUETUG, The College and University Eleven Thirty Users' Group, with an initial membership requirement of sharing at least one piece of software each year with the other members, ASCUE has a strong tradition of bringing its members together to pool their resources to help each other. It no longer requires members to share homegrown software, nor does it have ties to a particular hardware platform. However, ASCUE continues the tradition of sharing through its national conference held each June, its conference proceedings, its newsletter, and its sponsorship of NECC, the National Educational Computing Conference. ASCUE proudly affirms this tradition in its motto "Our Second Quarter Century of Resource Sharing".

ASCUE's ASCUE-L LISTSERVER

Subscribe by sending the email message SUBSCRIBE ASCUE-L yourname to listserv@purccvm. The listserver itself is ascue-l@purccvm.

NEED MORE INFORMATION?

Direct questions about the contents of the 1994 Conference to Mary V. Connolly, Program Chair, ASCUE '94, connolly@saintmarys.edu, Mathematics Department, Saint Mary's College, Notre Dame, IN 46556, 219-284-4497. Also, subscribe to ASCUE's listserver, ASCUE-L. Details about future conferences will be sent to the listserver.

"We hereby grant ERIC non-exclusive permission to reproduce this document."

Association of Small Computer Users in Education
ASCUE 1993-1994 BOARD MEMBERS
(years remaining in office)

PRESIDENT
F.P.(Rick) Huston, III (1 year)
Computer Services Division
University of South Carolina-Aiken
171 University Parkway
Aiken SC 29801
803-641-3422
rhuston@univscvm.csd.scarolina.edu

PROGRAM CHAIR
Mary V. Connolly (1 year)
Mathematics Department
Saint Mary's College
Notre Dame IN 46556
219-284-4497
connolly@saintmarys.edu

TREASURER
Thomas A. Pollack (1 year)
Associate Dean/Director of
Undergraduate Studies
Duquesne University
600 Forbes Avenue
Pittsburgh PA 15282-0104
412-396-1639
pollack@duq3.duq.edu

SECRETARY
Dagrun Bennett (2 years)
Computing Services
Franklin College
501 East Monroe
Franklin IN 46131
317-738-8150
bennett@minnie.franklincoll.edu

IMMEDIATE PAST PRESIDENT
Carl P. Singer (1 year)
Computing Services
DePauw University
Greencastle IN 46135
317-658-4294
runner@depauw.edu

BOARD MEMBER AT LARGE
Thomas E. Gusler (1 year)
Associate Provost for Adminstration
Clarion University
103 Carrier Hall
Clarion PA 16214
814-226-2676
tgusler@vaxa.clarion.edu

BOARD MEMBER AT LARGE
Gerald Ball (2 years)
Mars Hill College
P.O. Box 370
Mars Hill NC 28754
704-689-1242
gdball@rock.concert.net

PUBLIC RELATIONS
Don Armel (1 year)
School of Technology
Eastern Illinois University
Charleston IL 61920
217-581-2216
cfdja@eiu.edu

LIBRARIAN/HISTORIAN
Jack Cundiff (1 year)
Horry-Georgetown Technical College
P.O. Box 1966
Conway SC 29526
803-347-3186
r010125@univscvm.csd.scarolina.edu

Edited by
Rick Huston, University of South Carolina - Aiken
Don Armel, Eastern Illinois Univeristy

Table of Contents

About ASCUE 1
ASCUE-L listserver 1
Need More Information? 1
Board Members 2
Keynote Speaker 7
Eric Johnson
Pre-conference Seminars 7
IT Planning — Ronald Bleed
COMPEL — Lorinda L. Brader
Internet — William P. Wilson

Papers

Developing a Campus-wide Computer Ethics Policy 10
Lisa Brown, L. W. Colter, Trey Short
Illinois Wesleyan University

Computer Integration in Professional Education: Making Technology More Than a Convenience at a Regional Institution of Higher Education 23
Sue Buryanek
Wayne State College

Networking Everywhere: The Development of the Bentley Information Resource 26
David R. Callaghan
Bentley College

ERDA: A Model for An Intelligent Learning Environment For Abstract Data Types 34
Larry C. Christensen
Bill Hays
Brigham Young University

Adopting Client/Server Computing: A Model for Small Colleges 44
CDR Steven M. Conway
Mr. Barry M. Levine
United States Coast Guard Academy

Learning Efficacy and Student Reaction to Computer Assisted Instruction 48
Amit Dave
Ed Billmire
Life College

The Impact of a Networked, Multidisciplinary-use Computer Classroom on the Whole Campus of a Small Liberal Arts College 56
Kathy Decker
Clarke College

Growing Pains!!!! Where Do We Grow From Here???? 61
Planning the Growth of Computing Resources at Rhode Island College
Allyson B. Dickie
Rhode Island College

An Analysis of a Computer-Based Self Study Module: Student Reastions 71
William J. Gibbs
Donald Armel
Eastern Illinois University

Attitudes of CIS Graduates Toward Course-work Needed For Entry-Level Computing Positions 79
Gireesh K. Gupta
Belmont Abbey College

Auditing: One Answer to Meeting Faculty Development Computing Needs 91
Joan M. Hofmann
St. Joseph College

Mobile Computing at Grove City College 103
Frederick J. Jenny
Grove City College

Computer Applications in the Physics Curriculum 108
H. R. Kobraei
R. C. Etherton
Murray State University

Artificial Neural Networks for Undergraduates 111
Patrick J. C. Lamont
Western Illinois University

Tale of Composition Wizardry: A Teacher and His Student Relate Their Adventures in Virtual Reality 119
John LaRoe
Matt Roberds
University of Missouri - Kansas City

Internet Access, Navigation, and Resource Location: The Future is Here 128
Bill Z. Manaris
Christopher A. Jones
Fayetteville State University

Xplore As An Inexpensive Alternative 138
Carol A. Marinas
Barry University

Meeting the Theory Needs of Computer Science Majors in a Small Computer Science Program 143
Renee McCauley
Wintrhrop University

Equipping the Classroom Data Jack 151
Michael R. Meeker
University of Wisconsin Oshkosh

An Expert System Prototype: Copyright Law in a Multimedia Environment 159
William J. Moressi
Renée A. McCauley
Winthrop University

Personnel - The Key to Making It Work 174
Warren Murphy
Troy State University

Computer Competencies: Setting a Quality Standard 178
TQM Team #3
Janet Polansky
University of Wisconsin-Stout

Ethical Issues for Information Systems Professionals 191
Thomas A. Pollack
Duquesne University

Listserv Discussion Groups on Internet/Bitnet 199
James W. Reese, Ph.D.
Associate Professor of Economics
University of South Carolina, Spartanburg

Strategic Planning to Implement a LAN from Ground Zero 205
Do Yeong Shin
Shaw University

A Viable Alternative to Present Advising and Registration on the University of South Carolina at Spartanbrug Campus 210
Stuart Shough
Chuck Morton
University of South Carolina, Spartanburg

Where to Introduce C in the Small College Undergraduate Curriculum 212
Peter D. Smith
Saint Mary's College

Using Maple in the Calculus Sequence 220
Thomas Smith
Charles Stolze
Patrice Tiffany
Kathryn Weld
Manhattan College

Centralized Computer Direction Without Centralized Computer Control 228
Robin M. Snyder
Shenandoah University

Proactive Approaches to Information Systems and Computer Security 236
Robin M. Snyder
Shenandoah University

Computerizing on a Shoe String: With Lots of Prayer 243
Richard N. Stewart
Lutheran Theological Seminary at Philadelphia

Freshman Biology Majors Experience Authentic Learning 249
by Producing Their Own Lab Manual
Paul S. Tabor
Clarke College

Stimulating Simulations 253
Ken Taylor
Thomas More College

Moving Small Colleges into the Information Age: 276
Problems, Resources, and Strategies for Success
Dian Thompson
Edgar H. Thompson
Emory & Henry College
John Stuckey
Washington and Lee University

Design Issues in Computer-Based Instruction 294
Don Armel
William J. Gibbs
Eastern Illinois University

Impact of Telecommunications on Education: Some Serious Considerations 303
David Ayersman
Dusty Anderson
West Virginia University

CWIS Development from Beginning to End 317
Eric Gallagher
Christine Mathews
Hood College

Personal Computer Security 331
Gustav J. Stangline
Wilkes University

Connecting Students And Faculty 336
With The World Using The Internet!
Gustav J. Stangline
Wilkes University

Planning Effective Learning Experiences 341
Elizabeth Lancaster Braswell
Ariel Associates

Presenters Index 349

Keynote Speaker

Eric Johnson is the Editor of TEXT Technology: The Journal of Computer Text Processing, and he has written or edited nearly one hundred articles, monographs, and papers about computing. Six times he was the Director of the International Conference on Symbolic and Logical Computing and the editor of its proceedings. He has created commercial software (StrongWriter, a grammar and style checker) and production software used at his university and at other schools (a document indexing program, software to assist in creating academic schedules, a program to record and calculate basketball statistics, and accounting software). He has taught a graduate course in computer programming for the humanities via Internet. His Ph.D. degree from the University of Notre Dame is in nineteenth-century literature and literary criticism. He is Professor of English and Dean of the College of Liberal Arts at Dakota State University.

Preconference Seminars

Preconference Seminar 1
The Learning Action Plan: A New Approach to IT Planning for the 90s

Presented by:
Ronald Bleed, Vice Chancellor Information Technologies, Maricopa Community Colleges

This workshop will introduce participants to a new approach to strategic planning for information technology in higher education. This new approach, referred to as the Learning Action Plan, focuses on the institution's need to: (1) reexamine its basic business processes, (2) determine how information technology can best be used to address the business problems of the organization, (3) expand its alliances with new and different types of external organizations, (4) craft new "owner communities" for use of information technology, and (5) incorporate the principles and techniques of total quality management into its daily operations. This workshop will present a rationale for why a new approach to strategic planning is necessary for IT, and then, using a case study of the Maricopa Community Colleges, illustrate how other institutions can implement this new planning approach.

About the Presenter:
Ronald Bleed has been involved with college computer centers for 27 years and has been with Maricopa Community Colleges for 14 years. A popular workshop presenter at CAUSE, he brings to ASCUE an impressive background in strategic planning.

Preconference Seminar 2
Using COMPEL to Create Multimedia Presentations

Presented by:
Lorinda L. Brader, Faculty Computing Support Center Manager Academic Computing, East Carolina University

COMPEL is a brand new full–featured graphics presentation product released by Asymetrix, the producers of ToolBook. Using COMPEL, you can create and deliver shows that incorporate the latest in special effects and multimedia, including text, graphics, sound, animation, full–motion video overlays or digital video. With COMPEL's unique linking and navigation features, you can quickly respond to and interact with your audience. Just click on an object, button, work, or bullet to link to other slides, presentations, or applications. You can also press "TwinClick" to access an on–screen navigation panel that is unique to COMPEL. You can get the look you want for your message by selecting from over 100 templates included with COMPEL, or you can customize your own templates for use in your presentations. Then choose from a variety of slide and bullet transitions to keep your presentation interesting. You can create high–impact presentations that reflect your own style by including sound, animation, and video. Select from an extensive library of media clips that come with COMPEL, or create your own clips to include in your presentation. Since COMPEL supports Microsoft Video for Windows, you can include digital video clips in your presentations. In addition, COMPEL automatically generates overheads, handouts, speaker notes, and slides. And if you need to travel, COMPEL includes special utilities that compress, package, and check your files. Then you can show your presentation on any other machine using Windows and the runtime version of COMPEL. Whether you're about to create your first multimedia presentation, or you're an experienced presenter looking for a way to get your message across more effectively, this workshop on COMPEL can benefit you.

About the Presenter:
Lorinda Brader worked as an instructional courseware developer for Penn State for five years. The recipient of a number of awards for courseware, she is back at ASCUE by popular demand.

Preconference Seminar 3
The Essential Internet

Presented by:
William P. Wilson, Advanced Software Support Coordinator Computer Services,
Gettysburg College

Many of us are still getting our feet wet with the Internet, let alone learning about all the sources, information directories, etc. This workshop will introduce you to most of the tools that will allow you and members of your academic community to enjoy their interactions with the Internet and all it can provide. During the day we will talk about the following services, what you need to run them, and what you need to develop your own information centers on-site.

- Mail
- Interest Lists
- Usenet news
- Gopher
- Mosaic

This discussion will be non-technical, and we'll spend much of the time looking at various clients for Mac and PC platforms. We will spend some time discussing servers so you will see what's involved with setting up some of these services right on your own network. Some time will be reserved for people to explore for resources on the net.

About the Presenter:
Bill Wilson is well known for his work at Gettysburg. A former ASCUE president, he has been guiding ASCUE members into networking and the Internet over the past several years.

Developing a Campus-wide Computer Ethics Policy

Lisa Brown, L. W. Colter, Trey Short
Illinois Wesleyan University
P.O. Box 2900
Bloomington IL 61702-2900
309-556-3421
brownlj@vmd.cso.uiuc.edu

Introduction

As part of a campus-wide computerization planning effort, Illinois Wesleyan University realized it would be necessary to set in place rules of conduct, methods of monitoring conduct, and penalties for transgressions of these policies. Areas of particular concern were: the protection of academic freedom and privacy for faculty, compliance with software license agreements, and the problems associated with a campus dependent on distributed processing and desktop computing. The policy was developed by a faculty task-force and approved by the administration without revision. The present paper discusses the process for developing the policy and the means for implementation.

Background

Illinois Wesleyan University is a small, liberal arts university located in central Illinois. The University serves an undergraduate population of approximately 1800 students in the College of Liberal Arts and the Schools of Fine Arts, Music, Nursing, and Theatre Arts. The Carnegie Foundation reclassified Illinois Wesleyan recently from a Regional Comprehensive II to a Liberal Arts Baccalaureate I institution, largely on the basis of significant increases in the quality of the institution.

The Board of Trustees heads the University's structure. It is governed daily by its President, Provost/Dean of Faculty, and Associate Dean. Faculty committees relevant to campus computing include the Council on Policies and Procedures (CUPP - a campus-wide, representational committee). In addition, the Academic Computing Committee (an appointed committee); Illinois Wesleyan has institutional methods of dealing with misconduct for students and faculty; there is a standing Hearing Committee that works with the Personnel Council in cases of faculty misconduct; and the University has a Judicial Review Board for cases involving student misconduct.

Until recently, Illinois Wesleyan University had very little computing on campus. A nearby state university (Illinois State University) serviced the needs of Administrative computing. Faculty computing was not supported. The only academic computer lab contained twenty Tandy, dual floppy computers and three terminal hook-ups to the University of Illinois. There was and is no mainframe on the campus, although the

University supports administrative computing via the recent acquisition of an IBM AS/400.

In 1989, the University made a serious commitment to providing computing capabilities to students and faculty. The desktop computing population has grown to about 450 units. Each faculty member has a desktop system. The University renovated a building and created an open-access lab, two computerized classrooms with desktop units available at each student desk, and an interactive learning center to support multimedia courseware development and use. There are also a few special-purpose labs to support specific departmental needs. As noted above, administrative computing now is supported using an AS/400 system. Desktop units are approximately equally divided between Macintosh and PC clone machines. Several classrooms have computerized teacher's stations to support multimedia presentations. The Library has increased its connections with on-line searching facilities and the use of CD-ROM technology. In addition, the University is implementing an E-mail system using the campus digital phone lines and has recently become an Internet node.

The institution has benefitted from its lack of previous computing. There were no old machines to replace, no mainframes in which the University had sunk enormous amounts for support. Other than the connections to the AS/400, the entire campus uses desktop units. In the labs and classrooms, a Local Area Network (LAN) connects the computers. The Academic Computing staff supports all desktop units, manages the labs and classrooms, and performs maintenance and repair on all units.

One of the classrooms is a Writing Workshop. This is a Macintosh-based classroom with 20 student units. All freshmen expository writing students are taught to use these computers as part of their expository writing course. The other classroom is shared by computer science, business, and economics, and sociology. It contains 29 student IBM PC's.

The future of computing at Illinois Wesleyan University is very positive. The University is building a new science building with several computerized labs, several small research computers, and a building network.

Why Was a Policy Necessary?

With this explosion of hardware and software on campus, several new problems arose. When computers were located in one room, it was relatively easy to watch the users and ensure the copyright laws were followed, at least with regard to software. Since no computers were connected to any others, no one could do any damage to all the systems without considerable effort. Of course, it was also possible to monitor use and track problems when they occurred.

The new environment posed two major problems. One was the network integrity in the networked building from both malicious and accidental abuse from users. The other, what was happening across the campus in faculty and staff offices, and software copying. Illinois Wesleyan is not a University with an enormous software budget. Choices had to be made and software purchases were carefully considered. Without infringing on privacy rights and academic freedoms, the institution wanted to discourage its employees from participating in copyright infringement in any form.

For the University, there was also the concern for liability. Some institutions have had to institute policies as a result of criminal prosecutions and lawsuits. While the legal liability of an institution for activities of individual faculty, staff, and students is not completely clear, it is likely that the risk of such liability is reduced by reasonable efforts to control copying[1].

In addition to all these factors, software companies were becoming more aggressive in their attempts to discourage copyright infringement. It is not the purpose of this article to argue the extent of the copyright law, but it is widely acknowledged that software is protected and cannot be copied freely for classroom use or personal use, much less to spread around the institution[2].

How the Policy Was Developed

As the need for such a policy became clear, it was the Academic Computing Committee who decided to see the task through to completion. This committee had members drawn from the faculty and administration (including the Manager of Academic Computer Services) and was, therefore, an ideal spot for such discussions to occur. It was decided to appoint a small task force to create the policy which would then be reviewed, edited, and approved by the entire committee. The appointed task force consisted of a Philosophy professor (Dr. Larry Colter), a Computer Science professor (Dr. Lisa Brown), and a member of Academic Computer Services (Marilyn Barnes).

The task force compiled an extensive list of policies implemented at other institutions and used this information as a basis for the IWU policy. The task force determined some underlying principles on which it would base its policy:

1. The policy should be enforceable. Without enforcement, the policy would be useless and ignored. Therefore, the institution had to endorse the policy formally.

2. The policy should respect the ethics of professional colleagues. The policy was intended as a technique for the University to police itself and not as a method for harassing colleagues.

3. The policy had to respect faculty member's, staff's, and student's rights to privacy. So in much the same way as the University has no right to control what one creates with the use of a pen and paper but can protect itself if a faculty member uses that pen and paper to commit sexual harassment or to write threatening letters, it has no right to investigate the works created using the tools provided by

the institution (in this case, the hardware and software that comprise the computer system).

4. The policy had to respect the rights of the student users. Forced to use a facility to support classroom assignments, the students would not be coerced to signing an oath.

5. The policy should be legally viable and defend the rights of the institution.

In many ways, this policy was handled in a way similar to the University's sexual harassment policy. Such policies must be implemented by an institution and, while requiring faculty input and participation, are implemented and imposed by the administration for the benefit of the institution. This meant that the task force had to include the comments, where possible, of faculty members and committees, but that the policy itself had to be approved and implemented by the administration and the Board of Trustees.

The task force performed a search for information on such policies in institutions across the country. In looking at policies for approximately forty colleges and Universities, the committee noticed that some policies that sounded good would be unenforceable in a micro-computing environment. Many policies depended on the use of accounts and user identification to establish methods for tracking misuse and for enforcing the consequences mandated by the policy. Many institutions had very rigorous penalties, with dismissal from the institution a very real possibility in policies like that of the University of Michigan. Reading these policies gave the committee the sense that a balance had to be struck between the intimate atmosphere and sense of community encouraged at our campus and the increasingly likely possibility of a case a misuse occurring.

The committee did not confine its research to campuses within the United States or post-secondary institutions. Even public schools were beginning to cope with the problems of software copyrights[3]. Universities in other countries were also exploring the need to improve their policies given the increasing use of software and Internet at their institutions. In all, the committee reviewed well over 50 policies and procedures and participated in some of the Internet discussions concerning such policies.

The committee had one other concern. Not only was it clear that computing was growing on our campus, but because of the lack of a mainframe environment, there were still faculty members (actually increasing numbers of them) who were linking with the University of Illinois Computer Center to perform some of their research computing tasks. It was therefore important for the committee to include some acknowledgement of the University of Illinois Computer Use policy and support its implementation on our campus.

In drawing up the policy, the Task Force had another choice to make: Should the policy be strictly a legal document, or should it go beyond the strictly legal and create a policy which also emphasizes the moral rights and obligations of staff and students with respect to computer use? Illinois Wesleyan is the sort of institution in which matters of value, of fairness, justice, integrity, etc., lie squarely at the center of its mission. As an institution, we seek to promote and enhance the capacities for and commitment to acting as good citizens of a global community.

The Task Force decided, therefore, that our policy should reflect those institutional commitments, and so the policy that follows immediately below goes beyond the strictly legal issues and focuses on ethical issues as well.

The Policy

Illinois Wesleyan University is committed to the proposition that an academic institution is a community in which the ideal of honesty is to be fostered, encouraged, and achieved. Respect for the University, for one's fellow humans, and for their property - both real and intellectual - are therefore essential ingredients of that ideal, and the University expects of all its members that they exhibit such respect. The ideal of honesty is of course a moral ideal, and so the policy stated below will in some respects go beyond the mere requirements of the law.

Computing technology, because of its extremely volatile[4] nature, presents strong possibilities and hence temptation for misuse. It is doubly important, therefore, for all members of the University community to be aware of that fact and to be doubly committed to use such technology appropriately and to show the respect described above. Accordingly, and for the benefit of all members of the University, the information technology usage policy stated below is intended to make clear just what constitutes that respect, and all members of the University are expected, on pain of penalties described herein, to abide by this policy.

All users of the University computer facilities must agree to use the facilities legally, ethically, and in keeping with their intended use.

1 *System Integrity*

Actions taken by users which interfere with or alter the integrity of the University's computer system are improper. Such actions include unauthorized use of accounts, impersonation of other individuals in communications, attempts to capture or crack password, attempts to break encryption protocols, compromising privacy, destruction or alteration of data or programs belonging to other users, and attempts to steal or destroy software resident on campus computing facilities. It is improper to create worm or virus programs or conduct experiments to demonstrate computer facility vulnerabilities without prior permission of Academic Computer Services, or to create programs which disrupt or interfere with other users' computing processes. Users are responsible for damage caused by infected software they introduce into the system.

2 *Copyright Observance*

All users of University-owned computers are expected to abide by copyright laws and licensing agreements. No software should be loaded on any University computer in violation of licenses or laws. No user may copy, or attempt to copy, any proprietary or licensed software provided or installed by IWU.

The University recognizes its role in education for ethical behavior in the computer setting as well as elsewhere. To that end, the Manager of Academic Computer Services will provide, when requested, information about copyright and licensing issued to members of the University community. Said manager will not be liable for copyright or licensing infringements by and student, faculty, or staff member.

The central "fair use" concept of the 1976 copyright law allows borrowing of small amounts of printed, audio, or video materials for such uses as "criticism, comment, news reporting, teaching,... scholarship, or research" [Copyright Revision Act, p. 16]. The test of fair use addresses (1) the purpose and character of the abuse; (2) the nature of the work copied; (3) the proportional amount copied; and (4) market effect. Aside from legal issues, users should recognize that the violation of copyright laws with respect to software drives up prices, discourages vendors from offering education pricing, and makes the development of good software a risky investment of the developer's time.

3 *Privacy Rights*

The University respects every individual's right to privacy in the electronic forum and prohibits users of University computers, including personally owned computers linked via University telecommunications equipment to other systems such as the University of Illinois computer system, from violating such rights. Attempts to read another person's electronic mail, to access another's files, to access electronic records containing information concerning another person, or to use another person's password represent examples of violation of privacy rights.

4 *Courtesy*

Anuses of University-owned public access computers may result in the suspension of use privileges. Such abuses include (but are not limited to):

excessive use of paper,
making electronic mass mailings,
using University-owned computers for personal monetary gain (except as such use relates to professional development),
monopolizing machines,
and other similar or related abuses.

In addition, University-owned public access computer will not be used for games for other than educational purposes. In general, electronic mail is to be used for academic purposes only. Pornographic, threatening, or nuisance messages are violations of the user's pledge to use computing facilities ethically.

5 *Sanctions*

The University may take disciplinary and/or legal action against any individual who violates any computing policies, including temporary or permanent suspension of an individual's use privileges to all or part of the college computing facilities, temporary suspension from the University, or permanent separation therefrom. Student violations of academic honesty standards for classwork will be reported to the Office of the Associate Dean in accordance with regulations described in the student handbook.

6 *Liability*

Illinois Wesleyan University hereby expressly and explicitly disclaims any liability and/or responsibility for violations of the policy hereabove stated.

Thorny Issues

There were several topics that created considerable discussion. One of the easiest questions to handle was whether or not to allow the use of games. In the former environment, it was possible to allow controlled game use because the use of the computers could be monitored easily from a single vantage point. In the new facility, it was no longer easy to see what each student was doing. Since the computers were being provided to support academic instruction, the task force felt that students using the computers for that purpose should have priority over students using the computers for recreation. In this particular case, the solution, to not allow games unless used as a simulation or exercise for instructional purposes (such as a foreign language game), was easy to implement as well. The computers in question were all in a single building connected to a single network. No games were provided as part of the network software, so use was effectively prohibited.

There was considerable discussion concerning the use of the computer equipment for monetary gain. While no one felt the University should support someone running a business using the University computers, it was difficult, especially with regard to faculty use, to draw an appropriate line for practice. In this case, the task force used the same standards they might use for pencil and paper supplies provided by the institution. The university continued to support the use of tools provided by the institution in support of faculty research and development that was not funded by another agency. Computing time on other mainframes was also supported within reason. So the task force felt this standard could be used to support the writing of books and papers for publication, etc.

One of the techniques used by software companies to control copying is to provide a limited set of manuals. The task force felt this practice was an excellent way to help enforce the licensing agreements. Most software packages are too complicated to use without some form of manual. Academic Computer Services tracks each piece of software and works to make sure manuals are controlled.

Another area of considerable concern was academic dishonesty cases that occurred in the computer science department. students who would not have considered the possibility of copying someone else's term paper, were not getting the message that a program was also someone's intellectual and creative property. It was considered important to add some statement in support of the computer science department's efforts to correct this misconception and prevent unnecessary occurrences of plagiarism. The original policy had an attachment detailing the events that would be considered as academic dishonesty in the context of software, but this statement has been incorporated in the syllabi of the computer science courses instead.

Implementation

The committee devised, along with the policy, an implementation plan. The first stage of this plan was to obtain the approval of the Board of Trustees. To this end, the task force developed a rationale for the plan - essentially o three page position paper explaining the reasons for such a policy and some of the supporting evidence for the policy. In particular, the task force felt it important to further describe some of the wording of the policy (for example, using University-owned computers for personal monetary gain). This rationale and the policy, accompanied by a member of the Academic Computing Committee, were then sent to each of the major University Councils (Personnel, Curriculum, and CUPP) to request review and comments. The policy and rationale were also sent to the campus attorney, whose comments worked to improve the document and make it more legitimate legally. After the committee included all appropriate comments, the final draft of the policy was presented to the Board of Trustees for approval.

The second stage of the implementation plan was to include the policy in the faculty and student handbooks. The policy was presented to the faculty very successfully thanks to the previous work with each of the Councils and the work of the members of the Academic Computing committee in explaining the plan. It was considered exceptionally important that the Academic Computing Committee was evangelistic in working with the faculty on the plan. This was especially critical because the Academic Computing Committee has as its members faculty from most disciplines and those who are the most current with computing technology in their disciplines. They therefore commanded a certain respect and, if they were willing to live by these rules, it was felt the rest of the faculty would be too.

The third stage of implementation was to begin to make sure that every student was aware of the policy. The committee decided that registration was the best time to hand out the policy. It is at this time virtually every student on campus is seen in one location. Also, it is a place where students are handed a number of such materials. In order to be sure the students could not claim ignorance of the policy, the committee decided each should sign a sheet saying they had red the document and understood its meaning.

The Signature Sheet

Illinois Wesleyan University
Academic Computer Services
Computer Facilities Use Agreement

I, __________________________________, am being granted permission to use the academic computing facilities at Illinois Wesleyan University, including, but not limited to, microcomputer facilities in Buck Hall, Sheean Library, and Sherff Hall, telephone and network access, and associated peripherals for academic, non-commercial purposes. Permission is also being granted for use, on site, of licensed software.

I have been given, have read, and understand the terms of "Information Technology Usage Policy" which governs the use of such technology at Illinois Wesleyan. As part of that policy, I agree to abide by the licensing agreements between Illinois Wesleyan and the software licensor for software programs, databases, applications, word processing packages, etc. I acknowledge that such software is proprietary, subject to copyright laws, and not available to copy, transfer, or remove from the facilities.

I also agree to abide by the policies of the University of Illinois, or other institutions as applicable, when using facilities controlled by their policies.

I accept full responsibility for enduring that my use of the computing facilities does not interfere with other users or the proper functioning of dais facilities. I acknowledge the right of the Academic Computer Services personnel to inspect and remove, if necessary as a function of responsible system management, any files resident on computer equipment under their supervision.

The Manager of Academic Computer Services may take actions against any violation of this agreement. These actions may include fines for damages, revocation of usage rights at all campus computing facilities, and further disciplinary action as deemed appropriate by the All-University Judiciary Committee, as well as legal action by owners and licensors of proprietary software.

Signed____________________________________ Date ________________

It was during this phase of implementation that the committee discovered a small problem in its plans. These sheets were to be returned to the lab desk in the computing lab. However, the campus did not have any way of monitoring the use of the computers to prevent a student from obtaining access without signing the sheet or if they had misused the equipment. The manager of Academic Computer Services has, to this end, worked at securing a magnetic strip reader to read the magnetic strip on the back of the student identification cards. These strips were already used on campus to track students' meal ticket status and other information. To use a computer, a student had to turn in his/her identification card in exchange for a start-up disk. This point of contact was considered the appropriate time to check for the signature and authorization. The purchase of the card reader included the software required to track the student identification numbers.

The Academic Computing staff also installed virus and file protection software where it was most needed. In this case, the Macintosh computers seemed more susceptible to abuse and the Macintosh network less able to prevent such abuse. The PC clone side of the facility was supported using Novell Netware and had more capabilities for protecting files, restricting access, and isolating the user from the system.

In the Office

Academic Computer Services has attempted, where possible, to purchase site licenses of the software most frequently used on campus. Often these licenses are discounted for educational use. Although the University does not mandate a particular desktop platform, personnel are encouraged to use the standard word processing, spread sheet, and database software supported for each platform. In this way, the University reduces the risk of departments purchasing one copy and sharing it among several faculty members. Faculty members have been encouraged to know the policy and contact the Academic Computing Services personnel if they have questions.

Academic Computing also produces a newsletter. This newsletter has been to inform the faculty and staff of problems and cautions concerning ethical use of computers. The newsletter regularly runs articles defining terms and explaining common misconceptions concerning software. The public relations compaign has been of great service in increasing the support for the Computer Policy among the faculty and staff.

The policy was adopted in April of 1993 by the Board of Trustees of the University. The process of defining a policy was started by the members of the task force in the Fall of 1990. The long time from inception to completion was occasioned by a serious consideration for process. Where possible, the task force strove to include members of the affected communities. This long process has really shaped a new attitude on campus towards software piracy. Although it has really never been too difficult to persuade the user community that serious abuse is improper, there has been a certain Robin Hood attitude towards software piracy and computer hacking. The education of the community in the ramifications of such actions in higher prices, less frequent upgrades, etc. has helped change some of these perceptions.

Another step taken by the committee was to enlist the help of the computer science department in educating the students most likely to violate certain sections of the policy. With repeated support of ethical use of computers in the classroom (as well as modelling this behavior for their students), the computer science department has been effective in stopping some of the abuse before it occurs or causes serious damage. This department works closely with Academic Computer Services and helps identify users who are causing problems.

The Future

Illinois Wesleyan is about to enter a new phase of computing. Plans are under consideration to network portions of the campus. A new science building housing Biology, Chemistry, Computer Science, Math, Psychology, and Physics will be configured as the campus hub. The building itself will have an extensive local area network. The implementation of a network, with its possible connection to administrative computing facilities to help faculty with registration and other administrative tasks, brings with it a new set of potential abuses. It also brings a new set of potential solutions, since version numbers of software can be monitored and updated centrally. The University has planned to work with the software vendors and hardware vendors to ensure the security and confidentiality of the new system.

1. Green, Kenneth C. and Gilbert, Steven W.; "Software Piracy - Its Cost and the Consequences;" CHANGE; Jan/Feb 1987; pp. 47-49

2. Lytle, Susan S. and Hall, Hal W.; "Software, Libraries & the Copyright Law;" Library Journal; July 1985; pp. 33-39

3. Zakariya, Sally Banks; "Play Fair with publishers, and Put School Software Pirate in the Brig;" The American School Board Journal; March 1985, pp. 38-41

4. By "volatile," the university means accessible, replicable (hence stealable), and destroyable.

Computer Integration in Professional Education: Making Technology More Than a Convenience at a Regional Institution of Higher Education.

Sue Buryanek
Conn Library
Instructional Technology Center
Wayne State College
Wayne NE 68787
402-375-7382
sburyane@wscgate.wsc.edu

In examining the history of technological change in the U.S. educational system, we find that one device after another has been introduced into the schools, yet only a few have truly revolutionized the way educators teach. In the past as an industrial society, technology was designed to make work easier and more productive. One sees this philosophy perpetuated in the educational system; it is convenient to use computers for clerical tasks, grading, for drills, or for games. Today in a society which relies on information for success, teachers must look beyond the convenience and develop new ways to use technology to support their work and their vision of what education is about. Higher education offers the ideal forum in which to explore new ways of combining media for effective communication, teaching, and learning. Teacher Education divisions are challenged to prepare students who can adapt to the rapidly changing technologies so they will become competent teachers for the future. This presentation will focus on the integration of new technologies into the teacher education program at a small liberal arts college.

Background

Wayne State College is a regional public college and is part of a three-school state college system geographically positioned to serve rural Nebraska. It has an average enrollment of about 4,036 students during the regular session and 2,283 for May, June, and July sessions (WSC, 1993).

The Instructional Technology Center is located at Conn Library. It houses textbooks, curriculum guides, easy, juvenile, young adults books, K-adult audiovisual materials, audiovisual equipment, a satellite receiver and closed circuit system. With the development of the campus computer network and the library automation project in recent years, faculty have access to journal indexing systems on the CD-ROM tower, global library catalogs, a variety of software applications, and an internet connection. Faculty assign the appropriate applications to students in their classes.

Implementation

After completing a review of the literature and holding discussions with education faculty, we decided to follow a four step process -- know where you are, plan and build for the future, establish linkage, and exercise leadership by modeling effective use of new technology (Bosco, 1987). Our goal was to make students aware of the potential of technology in education, to develop their self-confidence through hands-on experience, and to enable them to explore ways technology could be incorporated into their teaching style and the management of their future classrooms.

Our computing capabilities have increased significantly in the last four years, so we needed to determine where we were as far as technology and preservice teaching were concerned. We wanted to identify the opportunities available through new technologies, but not allow the new devices to direct our efforts. Our short range plan involved restructuring the program by revising the library component of three classes -- Introduction to Teaching, Secondary Schools, and Instructional Media and Technology. Traditionally, each of these classes were randomly scheduled in the Instructional Technology Center for one or two fifty minute class sessions and students worked independently on exercises. We rescheduled the library sessions so they became a more natural part of the education courses. This enabled us to link information seeking behavior, the use of technology, and specific teaching methodologies as the semester progressed. With the exception of word processing which is applicable in all three courses, each of the three classes introduced different types of technology. In Introduction to Teaching, students concentrated on information seeking strategies using the journal indexing systems on the CD-ROM tower and the library computer catalog. In Instructional Media, they used a variety of computer software, the video camera, and traditional instructional technology. In Secondary Schools students explored satellite programming, utilized the nonlinear capabilities of the laser disc player, and were introduced to resources and listservers on the internet. These technologies enabled them to go beyond the classroom experience and interact with the real world. In the future, we hope to provide another link by identifying more innovative teachers in the schools who will supervise students in their pre-teaching experiences.

In order for the classroom environment to reflect reality, students must become active learners with the tools that are shaping their future. The laser disc player was a natural choice for meeting this objective. Two programs we liked for the laser disc were "Classroom Management: A Case Study Approach: and Classroom Discipline: A Simulation Approach" The laser disc player is easy to operate and the user has control over the program. It supports a nonlinear approach to learning. In our classes, the instructors gave fewer lectures and spent more time working with the small groups and individual students. Instructors and the information specialist used the computer and other equipment as they taught and provided basic instruction to the class as a whole. Tutorials were developed for each type of equipment. Students worked in pairs or small groups to practice their skills and to design projects. As discoveries were made, students and instructors would hypothesize about outcomes, tested their theories, and learned together; instead of serving as the deliverer of information, the instructor became a facilitator of learning. Utilizing this new methodology, we hoped to perpetuate the role of the instructor as a leader by modeling (step four).

This was the most difficult step because although we agreed that modeling was important, we had some doubts that our skills were adequate enough to serve as a model. However, modeling is very important to the preservice student. Deans, professor, computer coordinators, and other administrators were asked "how well do undergraduate programs in schools of education prepare preservice teacher for technology use in the classroom?" "Not well enough, " was the answer. "James Poirot, chair of the Department of Computer Education and Cognitive systems at the University of North Texas, Denton, says there are a lot of computer-resistant faculty, in general, at the undergraduate level. And if they don't use it in their classrooms, the students they are teaching will, by default, lose an important area of instruction--that is, instruction by example."(Bruder,1989) In order to counteract the effects of technophobes, we pooled our collective knowledge and supported each other as we tried new strategies. We hope to assist other faculty through a mentoring program.

Conclusion

Technological innovations often provide a more comfortable and convenient method for instructors to do what they have done in the past. In order for technology to go beyond convenience, new teaching methodologies must be developed which allow students to become active learners. Educators at Wayne State College used a four step process to facilitate the integration of information technology into their preservice teaching program in order to prepare students who can adapt to changing technologies and become competent teachers for the future. In this way, small liberal arts colleges are playing a special role in guiding the continuing integration of information technology into our live.

References

1993-94 General & Graduate Catalog. Wayne, Nebraska: Wayne State College, August 1993.

Bosco, James, et al. "The Challenge of Electronic Technologies for Colleges of Education." Journal of Teacher Education (Nov./Dec. 1987): 25-28.

Bruder, Isabelle. "Future Teachers: Are They Prepared?" Electronic Learning (Jan./Feb. 1989): 32-39.

Networking Everywhere: The Development of the Bentley Information Resource

David R. Callaghan
Bentley College
Waltham MA 02254
dcallagh@bentley.edu

Abstract

With a strategic direction set by the faculty, Bentley College's Information Systems Division set out to build a collection of computer services delivered by a high speed network to all on-campus spaces. With the college's reliance on tuition revenue, a $1.2 million project meant that grants and other donations would be necessary if the project were be completed before the technology was outdated. Two years after we started the implementation, the Bentley Information Resource provides everything from library catalogs, networked CD-ROMS, email and the Internet- all of it free of charge to everyone in the Bentley community.

The Student Computing Environment

Bentley College is a specialized, non-sectarian, private business college in the Boston area. It has 3200 full-time and 500 part-time undergraduates, and 300 full-time and 3500 part-time graduate students. It has a reputation for being a leader in the use of computer technology, having had Apple II labs as early as 1979 and as one of the first colleges in the nation to require students to have their own microcomputer (1985).

Over the years the college's PC requirement has varied, not only in the machine used, but the degree of requirement. In the 1990's only full-time undergraduates are required to show 'ready access' to a machine: graduate students and part-time students are encouraged to acquire one but are not required.

'Ready access' is currently defined as one student per machine for freshmen and transfers and no more than two students per machine for returning students. This policy is likely to change to one student - one machine requirement in the fall of 1995.

Students are allowed to choose between MS-DOS/Windows based machines or the Apple Macintosh. The requirement include portability so that it can be brought to class. Students are allowed to bring desktop units, provided they accept responsibility to locate a portable should a faculty member request one. The college has sold nothing but luggables and laptops since the inception of our program.

To ease the cash flow burden, the college provides a rental program. Students can rent a notebook for $250 per semester, $100 for the summer. While this has been a traditional offering, it is a considerable amount of work to distribute and recollect each year and it is likely that this option will be discontinued in the fall of 1995.

The wide variety of equipment on campus is important as it meant whatever network we installed, it had to be adaptable to a wide variety of platforms. This influenced our decisions nearly as much as the guidance we got from the faculty.

Faculty Direction

In 1989 a committee was formed to help define where the microcomputer program was going: in particular to decide on the type of machine the students would receive in the fall of 1990. The committee broadened its scope when it realized that the machine choice depended on the environment that the college could be expected to provide.

Since student needs change over four years and they aren't the same for all majors, the committee determined that the high power machine needed by some (CIS majors for example) would be much more than other students would need and would be overly expensive. In addition, there was no assurance that this machine would meet all the needs of the student four years later.

The committee concluded that we could expect the student's machine to meet a minimum of 75% of the student's computing need and that the college would provide additional capability where needed. These special needs would be met either through labs or preferably, by expanding the campus network to all residence halls.

The vision defined communication as the next power application, likely to exceed that of spreadsheets (very important for business students) and possibly even word processing. Network expansion would enhance communication with faculty, other students and the world. The major barrier was not PC hardware or software, but the links to tie them all together.

Beyond communication, file serving of expensive applications and access to more extensive databases were expected. With the advent of remote control software (Carbon Copy and PC Anywhere), we hoped the network could provide access to more powerful PCs over the net. Since the dual Mac/PC platform decision was being made at the same time, it was even hoped that cross platform remote control would be possible (PC Anywhere had a Mac Client for controlling PCs at the time).

A key element of all this was that the system be easy to learn and use. All services were to be accessible without having to know where the service was located, only that it was somewhere 'on the wire,' and that a single username/password identifier was all that was needed.

Network Configuration

At the time the faculty was providing its direction, the Information Services Division had embarked on a couple of projects to upgrade facilities. First was the replacement of a Prime minicomputer with a VAX. Prime, like other minicomputer manufacturers at that time, was rapidly losing market share and it did not look like they would be continuing in business much longer. Several faculty and staff had worked at institutions with VAXes and knew that the email system provided by DEC had the power to transform the communication for the institution.

The second project underway was to replace proprietary 5Mbps ethernet with an 802.3 standard 10Mbps system. The initial plan was to replace existing terminal servers with faster terminal servers for faculty and administration and to move the older, proprietary equipment to the dorms.

As the goals of the faculty were taken into account, it became clear that greater speeds than asynchronous would be necessary to achieve the file and CPU serving. If the students were to get ethernet, then the faculty would need it as well.

When the VAX was installed, a number of policies were developed. To remove barriers to email use, all faculty and students would receive an account on the system without asking and without charge. Administrative users, who had their own VAX, could have an account on the academic system as well, if they requested it. Many did because the academic VAX provided access to Bitnet and also allowed students to send email to administrators (for security reasons, students cannot mail directly to the administrative machine). This meant that there would be some 8000+ users when the entire community was given an account.

The Bentley Information Resource Project

The wiring of the entire academic community with ethernet and providing services 'through the wire' became known as the Bentley Information Resource. It was a joint effort of three departments in Information Services Division: Academic Computing Services; Telecommunications and Network Services; and Systems Management and Operations.

After the initial faculty committee was disbanded, new committees were created to oversee the installation of the VAX and the new network. There was some faculty and student participation, but with the direction having been set by the earlier faculty committee, most of the decisions left were of a technical nature. We found that faculty and student attendance at weekly technical meetings dwindled and soon disappeared. The committee continued to keep key faculty, particularly those in out Computer Information Systems department, informed via minutes of the meetings and with personal visits.

Network Choices

We began our search for a network operating system by gathering information from every conceivable source. The faculty's desire for a single username and password, meant tight integration with our existing VAX which we expected to be around for some years to come. It was a critical component of our computing strategy as it provided statistical and database packages, was our email hub, and was the gateway to Bitnet, and later the Internet.

Novell, while obviously a popular choice, could not meet our needs. We wanted to provide each user with personal storage space on the network server, just like we did with the VAX, to parallel a true business environment. With version 3.x having a maximum of 1000 accounts per server (200 realistically), it would require too many servers for our 8000 users.

There was also a concern on how to manage the duplicate accounts on the VAX and the servers. Even if we could develop a way to propagate the VAX usernames to non-VAX servers, we still would have to educate users on which server their files were kept. Novell servers might serve faculty in departments well, but not the students who cross departments. It would be difficult to communicate to students the different locations for the own files and files the faculty might provide for them without them learning far more about networks than we thought they should. Since most of these services were already provided on the VAX in a form already understood by the community, it seemed Novell, like other network systems (Banyan) would require a lot of unnecessary duplication and complexity.

Our choice came down to Pacer and Pathworks, both of which were VAX based. These systems could take advantage of the account management, backup, recovery and systems management we were already doing. The incremental management of the server software was much simpler than the development and maintenance of multiple servers with a different operating system.

We were quite taken by Pacer as it integrated the Mac, DOS, and VAX platforms. It also had a very nice licensing arrangement based on simultaneous users on the server. Client software was free and the server could be acquired in 20, 50, or 100 simultaneous user increments.

Pacer had its problems however. It was primarily a Mac server product and was based on the Appletalk protocol. Upon experimentation we found Appletalk kludgy on the DOS platforms and very slow. In demonstrations with faculty we found the acceptance low as the Appletalk basis seemed incongruous with DOS majority.

We did use what we learned from Pacer in our discussions with DEC regarding Pathworks. The Pathworks server software was covered under the CSLG host licensing agreement for colleges, but like many other network operating systems, there was a license charge for each client. At the time, that charge was $400 per station. With 2500 student machines on campus, 100 in labs, and upwards of 300 faculty and support staff, licensing would cost well over $1M. Our DEC salesman's eye's were swimming in anticipation.

We explained that since we expected students to share 2-5 machines per network drop in the dorms and that faculty/staff to be more likely to use the net during the day while students used it at night, a simultaneous user license made more sense to us. We were unable to get them to change their licensing structure but we did get very favorable pricing included in a grant from them for the upper campus network replacement.

Planning for the network expansion to the dorms began during the summer of 1991. The broadband backbone cable was already in the ground, as a result of negotiations with the cable TV company which pulled it when they brought cable to the dorms. We still had to pull wiring to 900+ rooms, and provide hubs, bridges, and ethermodems to connect the room wiring to the broadband. With all this hardware, we also needed net management hardware and software to keep it all working.

Wiring alone was not enough. We also wanted to provide enhanced services which would require upgrading our VAX, adding disk space, acquire a CD-ROM server, purchase software for bibliographic searches and add an Internet link to our net. What were weren't sure of was how student would connect their computers to the net.

As noted before, students predominantly have notebook computers. Virtually none of them support internal ethernet, so an external adapter was required. To avoid the support nightmare of multiple ethernet adapters if the students were to provide their own, Xircom adapters were selected for all student PCs, desktops and notebooks alike. For the Macs, Asante SCSI adapters were the most cost effective.

The issue became who would pay for the Xircoms and Asantes, the students or the college. Even with educational discounts off of $400 list prices, it would be very expensive to ask of students particularly since they were already paying for the purchase or rental of a computer. We were concerned that if the adapter purchase were optional we would have low absorption and not achieve our ubiquitous objective.

Our preference was to have the college buy the adapters and loan or rent them to students at a minimal fee. Since we already had a computer rental program in place, the systems and people were there to handle this type of distribution, provided the college could afford the cash outlay.

Grants

While we had developed a plan on what we would do to create the Bentley Information Resource, funding it was still an issue. The administration was committed to the project and was planning to fund it over several years via the capital budget. We were concerned that by the time it was fully rolled out, it would be out of date and would only have to be replaced. Still, getting some funding was better than none and so we forged ahead.

In the fall of 1991 the Institutional Advancement (fund raising) department notified us that they were pursuing a local foundation to become one of the early donors to our new capital campaign. They were looking for projects that might interest the foundation whose goal was to reduce the cost of education to the students.

Three projects were proposed: one for expansion of library services; one for increased media services, primarily video; and the Bentley Information Resource. In a meeting with the son of the foundation founder it became clear he had a personal interest in computers and the network project was intriguing to him. We stressed that this project was going forward regardless and that any donations received from them would replace tuition dollars which would have been used.

We were quite pleased when the foundation notified us that it was giving $300,000 immediately and providing another $210,000 one-for-one challenge grant. The project was going to go much faster than we had hoped. We knew that having one grant would make it much easier to get others.

We again approached DEC since we would need more hubs, bridges and ethermodems for the dorms. We included a VAX server upgrade and Infoserver CD-ROM tower as well. We were pleasantly surprised when they came through with a grant at a higher discount than we had ever received in the past.

We contacted Xircom and found that they had an educational program for refurbished demo equipment at about 1/2 the list price of new. We were all set to order the units when we were notified that the chairman of the Accountancy Department and the Vice President of Marketing at Xircom were childhood buddies. The next thing we knew we were flying to California to make a presentation on our project after which Xircom gave us an even greater discount on the already low educational price.

At this point we had lots of equipment grants (discounts) but limited cash funds. The original $300,000 was already earmarked for the DEC equipment and wiring of the dorms. We didn't have enough left to buy all the Xircoms or Asantes we wanted to provide to students.

The Institutional Advancement folks were having a difficult time explaining the project to potential donors. In December 1991 they decided to make a video to show what we had in mind. Done by professionals, the six minute clip included demos of what we had accomplished up to that point, interviews with students, the library director, the VP of IS, and even the college President all saying how important this project was to the future of the college.

Within three months not only was the remaining $210,000 was raised, but an additional $30,000 was earmarked for the project! The video was a proven success and we had enough to complete what had originally intended and more.

Installation

While the money raising was going on we were installing where we could. In the summer of 1992 we started with faculty offices, choosing departments which would make good use of it without causing undue burdens (the CIS department already had a Novell LAN). In the fall offered the network connections in graduate housing few people took us up on the offer to come down and install it for them. In hindsight, they did not understand what is was we were offering. Between the low acceptance rate and the low fund raising success of the time, we were getting discouraged.

This disappointment was quick to fade however when an RA in the undergraduate housing asked to have it installed on his floor. When the students returned in the spring of 1993 we started our pilot test in some undergraduate housing, doing one floor at a time. The installations, all done at night, involved giving a one hour group training session/demo where the hardware was distributed, followed by software installation in the student's room.

The pilot installations occurred up to spring break when we found student interest in the network disappeared in favor of final exams and the upcoming summer. By the end of the pilot, we had installed in 110 rooms and on approximately 200 machines.

One thing we quickly learned was that we could not take this installation approach for the entire campus. We were spending two to three nights a week there, in addition to the day work we had to do. What we did discover was that we could mass produce the network installation and it was robust - there were very few failures either upon initial installation or in subsequent use.

Since the video worked so well for fund raising, we decided that for the fall we would produce one to replace the training session and show it over the college's cable TV channel. We would then distribute hardware via the same mechanism we used for rental computers. The only issue was how to distribute the software.

We decided to have the network software pre-installed on all machines at our annual PC distribution. Since we distribute roughly 1000 machines each fall (half rentals, half purchases), we knew we could cover almost half of our student population in one step. When we started the Xircom distribution in the third week of September, we informed students via flyers passed out by their RAs, how to determine if they had the software or not. If the software was not on their machine they were asked to bring their computer up to the distribution facility where the software would be installed within 24 hours.

We still had the issue of desktop units and the support of network problems, many of which required on-site visits. Since students only seem to be available for this type of support at night, we hired a student who showed great interest in the project. This, like the video, proved to be a major success as he tends to keep the same hours as the students he serves, yet he devotes less than 10 hours per week to troubleshooting.

Current Status

As of April 1993 we have distributed over 600 Xircom adapters, 100 Asante adapters to students. We also have over 150 faculty and staff on Pathworks and all classrooms have Pathworks and asynchronous connections. We have recently installed a second (redundant) Pathworks server to improve reliability and responsiveness and used the network to update the client software to reference this new device.

We have numerous file servers available: one offering publicly accessible software; another which is a personal 'drive' for each user which points to the same location as their VAX 'home' directory; departments are each given an areas where faculty and staff can share data and programs; a 'class directory' area where faculty can place files for students to retrieve; and most importantly, the 'virtual lab', where all software formerly available only in central clusters, is now available anywhere on the net.

We also offer central print servers, networked CD-ROMs (14 platters at the moment), have added the Internet, and expanded the library catalog system to include newspaper and magazine indexes. Most services are accessible from PC or Mac and many can even be accessed from off campus via our modem pool.

In less than two years we have transformed the computing infrastructure from a standalone PC basis with limited asynchronous connections, to a ethernet based environment with multiple new services. All with no increase in staff.

Future Network Services

Our services continue to expand. We are continuing expansion in faculty departments and specialized labs. The CIS Department is developing a Lotus Notes server that they want to offer to the college community. New Internet and CD-ROM offerings are coming on-line and will be looking to upgrade to version 5.0 of Pathworks over the summer.

As a result of the extra money received from the fund raising, we are now investigating asynchronous Pathworks so that all students, regardless of whether they live on campus or not, will have access to exactly the same services. We hope to have this available by the fall of 1995.

Conclusions

The development of the Bentley Information Resource was intensive of money, time, and talent. We are extremely grateful for the grants we received: in 1990 from DEC for a VAX 6410; in 1991 from DEC for the upper campus network replacement; and in 1992/93 from our local educational foundation, DEC, Xircom, and the many generous alumni and friends who funded the Bentley Information Resource project.

We must also recognize the staff who spent many, many overtime hours developing software, working with students in the dorms, and troubleshooting problems. They spent the time and showed their incredible knowledge of computers and networks in search of what they knew what a good cause. It could not have been accomplished without them.

What we learned can be used by others. Grants can be obtained for dormitory networks provided you have done your homework and have a solid strategy of what you want to do. We found the videos, both for fund raising and for training extremely helpful. The pilot test was important in understanding how the user community would interact with the system. Finally, student support staff who can be available when students want the help, will ease the burden on already over stressed user support staff.

We consider the Bentley Information Resource an overwhelming success. The server processor capacity was quadrupled from spring 1993 to fall 1993 and we buried it at 100% of capacity by December. It appears that the ubiquitous network and ready off campus access has increased the use of computing in the curriculum, particularly with email. Growth from fall to spring 1994 seemed to be doubling again so a second processor was installed over spring break. At the same time, our modem pool has grown 40% in the last year to service the off campus user.

Clearly, there must be something of value for this dramatic a shift in demand for computing resources. We seem to have given them what they want and they have come ... and come ... and come ... What more can we ask for?

ERDA: A Model for An Intelligent Learning Environment For Abstract Data Types

Larry C. Christensen, Bill Hays
Computer Science Department
Brigham Young University
Provo UT 84602
801-378-5670
lchris@cs.byu.edu

Abstract

Educational Research in Data Abstractions (ERDA) is a research project in the Brigham Young University Computer Science Department. Our goal is to gain new insights into methods of Computer Aided Instruction (CAI) and to improve the instruction of abstract data types in computer science. The project is creating tutoring modules to supplement CS 2 lectures by emphasizing algorithms graphically, using animated presentations with computer software. This paper describes the ERDA model.

Keywords: Computer Aided Instruction, Abstract Data Types, Intelligent Tutoring, Knowledge Representation, Hyper-text, Object-oriented Relational Model.

Introduction

Educational Research in Data Abstractions (ERDA) is a computer science project to provide animated graphical representation of data structures and their associated algorithms. The project's namesake, ERDA, is a mythical character that appears in an opera by Wagner. The character represents a powerful, all knowing sorceress, something we would like in our computer tutor. In the project's twenty lessons, ERDA is the mentor to the student seeking knowledge and encourages the student with meaningful interaction, and graphical visual representations.

The project is in its second year and has completed 14 of 20 lesson modules normally found in CS2, the second computer science course. While ERDA's central focus is to supplement CS 2 lectures and textual information, the ERDA lesson content is not text dependent. The project strives to show the behavior of Abstract Data Types (ADTs), and various algorithms, by emphasizing graphically based, animated presentations. Coded examples use the Object Oriented approach and strive to demonstrate good software engineering techniques such as modularity, low coupling, and high cohesion.

From the beginning of our research, ERDA team members have documented and implemented standards used in the production of CAI. Many of these standards came from the FRODO project, a CAI project that delivers all instruction for a first year programming course[Christensen,1986]. Frodo has delivered computer instruction to more than three thousand students and the standards have evolved over the last five years. These standards permeate every lesson and provide consistent structure to the lessons while still allowing individual originality. Establishing standards allow other CAI projects to use the ERDA framework and follow the ERDA defined standards.

A blend of semantic networks and hypertext features captures much of the domain information in beginning computer science courses. The features from semantic nets and hypertext gives representation adequacy and query flexibility. The blend of semantic nets and hypertext provide structure (such as links and nodes) for both ERDA's presentation and query.

THE Semantic Net and Hypertext Features of ERDA

Vannevar Bush first introduced the ideas of hypertext in 1945 [Bush,1945]. Bush observed that the human mind does not structure concepts in a linear fashion but in an associative fashion. The idea of associated items, linked together, is the foundation of hypertext.

Semantic networks show interrelationships in the knowledge domain under study. A semantic network consists of objects, concepts, or situations called nodes and links between the nodes.

The six links used in the ERDA tutoring system are Generalization-Specialization (ISA), EXAMPLE, SIMILAR, WHOLE-PART, USES, and ATTRIBUTE. The following describes each of these links.

The Generalization-Specialization link is the ISA link since it implies that one concept is a special kind of another concept. This link is identified in Figure 1 with the "I" symbol.

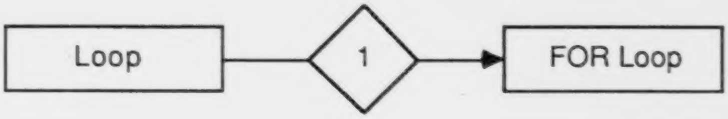


Figure 1
A Generalization-Specialization Link

An example of an ISA link is the association between the concept loop and the concept FOR loop. Read Figure 1 as "a Loop is a Generalization of a For Loop" or "a For Loop is a Specialization of a Loop or "a FOR Loop ISA Loop".

An EXAMPLE link connects a concept that is a specific instance of another concept. An EXAMPLE link is identified in Figure 2 with the "E" symbol.

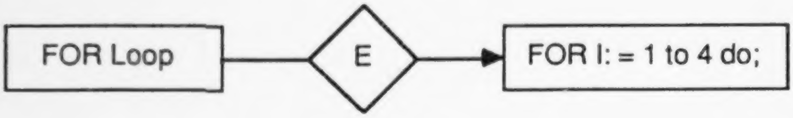


Figure 2
An Example Link

The concept "FOR I:= 1 to 4 DO;" is an example (or an instance) of the concept of a FOR loop. Figure 2 is read "a FOR Loop has Example For I:=1 to 4 DO;" or "For I:=1 to 4 Do; is an example of a FOR Loop."

The Similar link is identified in Figure 1 with the "S" symbol. An example of a Similar link is the relationship between a FOR loop and a WHILE loop since they are both closely related. Figure 3 is read "a FOR Loop is Similar to a WHILE Loop" or "a WHILE Loop is similar to a FOR Loop."

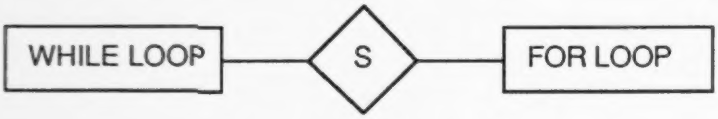


Figure 3
A Similar Link

Notice the Similar link is not drawn using an arrow. A Similar Link is a bi-directional link since the relationship between the concepts goes both ways.

The WHOLE-PART link is the aggregation link and is a concept comprised of other concepts. The WHOLE-PART link is identified in Figure 1 with the "W" symbol. An example of a WHOLE-PART link is the concept of a FOR loop with control variables, an initializer, a bound, and a statement block. Figure 4 is read "a FOR Loop is a WHOLE-PART of loop control variable and an initializer and a bound and a statement block" or "a loop control variable is a part of a FOR Loop" and " an initializer is a Part of a FOR Loop" and " a bound is a Part of a FOR Loop" and "a statement block is a Part of a FOR Loop."

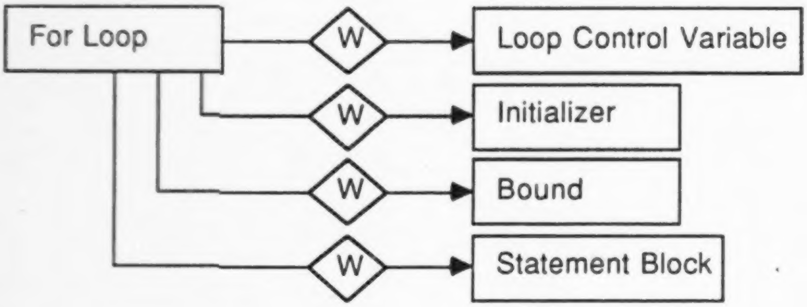


Figure 4
A Whole-Part Link

An Uses link connects a concept that uses another concept. The example of a Uses link is the concept that an array uses a FOR loop to do the function of initialization. Figure 5 is read "an Array Uses a FOR Loop" or "a FOR Loop is used by an Array."

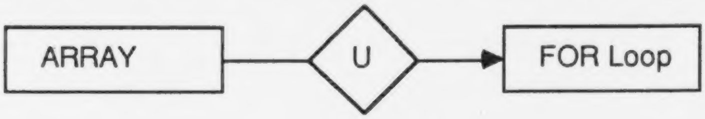


Figure 5
A Uses Link

The ATTRIBUTE link associates a concept that is a characteristic orrestriction on the connected concept. The Attribute link is identified in Figure 6 with the "A" symbol.

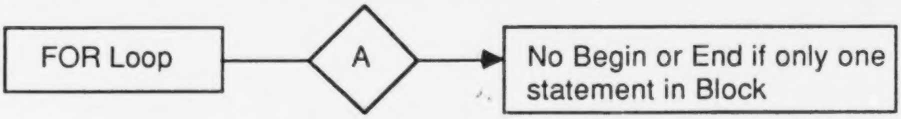


Figure 6
An Attribute Link

An example of an ATTRIBUTE link is that a FOR loop in the Pascal programming language has an attribute of no Begin and End when there is only one statement in the statement block. Figure 6 is read "a FOR Loop has Attribute No Begin or End if only one statement in block" or "No Begin or End if only one statement block is an Attribute of a FOR Loop."

Tutoring Model

The tutoring model for ERDA is centered around four parts; first, a dialogue between the student and the system, second, the creation of a student model and system model, third, an inferencing scheme to process the models, and last, a correction method for overcoming misunderstandings.

Romiszowski identified three criteria that separate an information system from an instructional system [Romiszowski,1990]. First, the system should allow students to practice their knowledge. Second, the system should allow students to practice their knowledge. Third, the system should monitor progress toward the goal. We follow his advice with the design of ERDA and explain features of the system.

System-Student Dialogue

Dialogue with a student consists of presenting information directed to a learning goal and acquiring information about the student status. When correction strategies for misunderstandings are needed, ERDA involves the student in the diagnostic and finally individualizes a remedial strategy. The dialogue is based on a mapping of the concepts interconnected with the six specialized links.

The concepts are organized as suggested by Novack and Gowin in their book [Novak & Gowin, 1984]. Concepts are represented in Figure 7 by rectangles with labels and links are represented by diamonds to identify link type.

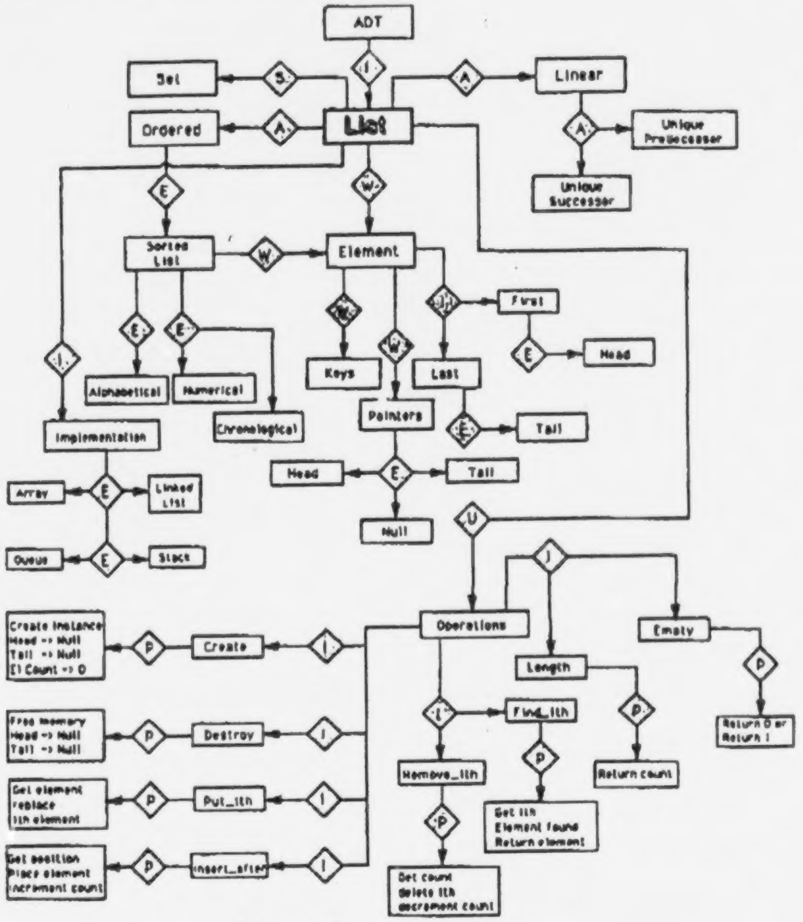

ADT
I
Set
S
A
Linear
Ordered
A
List
A
Unique Predecessor
Unique Successor
E
W
Sorted List
W
Element
E
E
W
W
J
First
E
Head
I
Alphabetical
Numerical
Chronological
Keys
Last
Pointers
E
Tail
Implementation
Head
E
Tail
Array
E
Linked List
Null
U
Queue
E
Stack
J
Create instance
Head -> Null
Tail -> Null
El Count -> 0
P
Create
I
Operations
Empty
Length
P
Free memory
Head -> Null
Tail -> Null
P
Destroy
I
I
Find_ith
P
Return 0 or
Return 1
Remove_ith
P
Return count
Get element
replace
ith element
P
Put_ith
I
P
Get ith
Element found
Return element
Get position
Place element
increment count
P
insert_after
I
Get count
delete ith
decrement count

Figure 7
Concept Mapping for Lists

ERDA begins a dialogue with the student to determine the student's knowledge of a concept in computer science. The system identifies the concept the student needs help with, then questions are presented for each of the links attached to that concept. The student chooses the link he does not understand. If for example, the student chooses "How do I insert a node in a binary tree?", then ERDA finds the nearest mastered node. This node might be the concept entitled "Binary Tree." ERDA presents the correctional text associated with the USES link between the concepts "Binary Tree" and "Binary Tree Operations". The correctional text associated with the ISA link that connects the concepts "Binary Tree Operations" and "Inserting a node" is then displayed.

If a student wishes to receive a lesson on a particular concept, then he activates a tutorial session. The basic design of the lessons follows a book metaphor. As with a book, the direction a student travels within a lesson is guided by the student: forward, backward, or non-sequentially. We have divided each lesson into chapters and concepts. Concepts are divided into the following types of frames: presentation, interaction, and examples. Presentation frames define and present terminology, algorithms, and processes.

The main method of presentation is through animated graphics. A moving picture is worth a thousand words. Interaction frames allow the student to move objects in the presentation window to show their understanding of a concept or to practice a process, such as stepping through an algorithm. Interactions may be simple, with only one movement or complex with exacting performance and spanning multiple frames. Finally, code examples are shown as graphically as possible to help the students synthesize the concepts and move toward implementation of abstract data types.

The interaction frames allow the system to gather information about the student and to assess the student's understanding of a particular concept. If a student is struggling with a particular concept, then a possible reason for their difficulty could be how the concept is linked with other concepts.

ERDA capitalizes on student expertise by first presenting a conceptual map of the concepts with its links and then asking the student questions relating to the links.

For example, in Figure 8, if the student did not understand the concept of a FOR loop, the questions listed would be:

A) Do you understand that a FOR Loop IS A Loop?
B) Do you understand that "For I:=1 to 4 Do;" is an EXAMPLE of a For loop?
C) Do you understand how an Array USES a For Loop?
D) Do you understand that a loop control variable is a PART of a For loop?
E) Do you understand how a While Loop is SIMILAR to a For loop?
F) Do you understand how "No begin/end if one statement in block" is an ATTRIBUTE of a For loop?

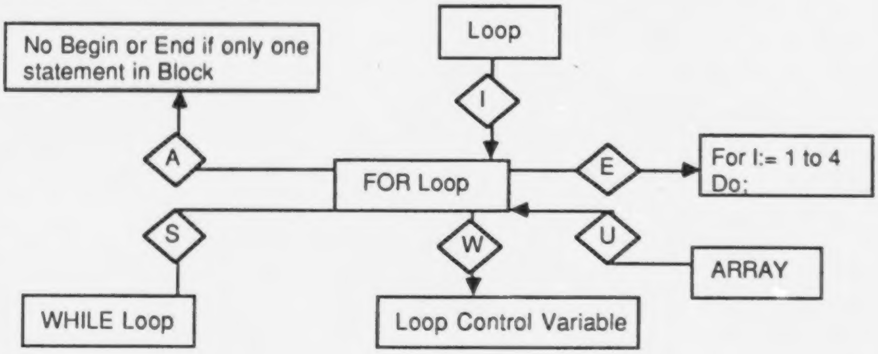


Figure 8
Concept Map for the concept of a FOR Loop

Since there are six links attached to the "For Loop" concept in the concept map, there are six questions listed. Notice that the wording of each question is closely related to the type of link that it represents. Hopefully, the student recognizes that he does not understand the contents of one of the questions and then identifies the question (link) to the computer system.

Strategy for Correcting Misunderstandings

If the student is not able to choose the link creating the misunderstanding, then ERDA begins a searching process to find the source of misunderstanding. The error exists somewhere on the path from a mastered concept to the problem concept. The algorithm to accomplish this correcting strategy is provided in Figure 9. The strategy is to move the student from a familiar concept to a concept that is difficult to understand. A concept is mastered if the student understands the links attached to that concept.

```
Test student on concept
Get solution
WHILE incorrect solution
    List question for each link attached to concept
    Get student's choice of link
    Find nearest Mastered concept
    If Same link is problem then
        Change Mastered concept to Not Mastered
        Find next nearest Mastered concept
    Display correctional text for each link in path to Mastered concept
    Retest student on concept
    Get solution
END
```

Figure 9
Algorithm for Correction Strategy

If a student can answer all the questions about a concept but not able to solve problems then the student's mastered concepts may not be mastered. If there are still problems with understanding the concept, then missing concepts and missing links are probably causing the misunderstandings. A different level of granularity resolves this problem.

A Typical Student Session With ERDA's Lessons

When beginning the Erda System, students receives the Introduction Lesson where they meet the mythical figure ERDA. Here, a conversation takes place between a surrogate student and ERDA. The student is asking for knowledge of Data Abstractions and Algorithms. ERDA sets them on the path to this knowledge, by directing them to begin the lessons. This introduction may be omitted on subsequent interactions.

Because the organization of the ERDA lessons is non-linear, a student may choose any lesson from the opening menu. If the student chooses the lesson on Sorting then they select a specific type of sort from the secondary menu. If the second sort, Selection Sort, is chosen, then a Title Screen appears, "Selection Sort, Chapter 2." A list of basic topics appears below the title. At this point, the student may evaluate their readiness for the lesson, continue with the lesson, go directly to the concepts with their links, or even select another lesson.

The student may tailor the flavor of their lesson at anytime. By clicking on the Settings Button, the student may select the language with which they wish to view code examples, activate or deactivate all or some of the three types of lesson concepts (Presentation, Interaction, Example), set animation speed and adjust the sound volume. In this simulated walk-through, the student chooses C++ as the code language. They are unfamiliar with the Selection Sort and request Presentations, Interactions, and Examples.

Upon beginning the Selection Sort Lesson, the student traverses the Presentation Frames. These frames define terminology and give a graphical demonstration of the sort. An example list of un-sorted elements appears with a pseudo-code algorithm.

The next type of frames that the student observes are the Interaction frames. The student demonstrates their understanding of the Selection Sort by dragging on the computer screen, with the mouse, the unsorted elements to a sorted position. If a mistake is made, then the move is undone and clues are given to aide the student. As in the Presentation frames, the use of color isolates sorted and unsorted portions of the list.

The final section of the lesson is the coded Example Frames. Because the student has chosen C++ as the example language, the sorting algorithm appears in this code. Where before the algorithm appeared in English pseudo-code, now C++ code is presented. A graphical animation of the sort appears with the code. When the student completes the Example, they should be able to implement a Selection Sort in C++.

The lesson closes with an Ending Title Screen. Again, the lesson lists the concepts. If the student is still unclear about the concepts, then the students begin the system-student dialogue. Nowthe student identifies the concept they are having difficulty understanding. The system now presents the links with their associated questions.

Conclusion

Erda, Educational Research in Data Abstractions, is a project to provide students graphical, animated representation of algorithms used in the CS 2 course. This research is using object-oriented techniques, concept mapping for the analysis, design and implementation of an intelligent tutorial system. This research includes the development of an user interface template that is highly portable, building an accurate student learning model, and a correction strategy for student mistakes.

Now beginning the second year of the project, Erda has moved into the testing and useability phase. Completed modules are currently used by CS 2 students. The graphical representation of information, the animated graphical definitions, the interactions and examples provided by ERDA, are providing meaningful support to the understanding of algorithms. Erda is allowing the user full control of movement, exploration, and interaction of their learning environment.

ERDA has given a new perspective to problems that are researched in Artificial Intellegence and Intelligent Tutoring Systems; first, a natural language interpreter would make the dialogue even more user friendly. Second, different levels of granularity are beneficial to students wanting a wider range of help in learning new concepts. Third, research is needed to determine if the concept has no way of knowing if all related concepts are represented in the map.

Finally, an attractive innovation of the project is the portable feature of the interface template. The metaphor of a book is valid for many applications and the features of paging, glossary, and indexing are independent of computer science concepts.

Bibliography

Bush, Vannevar "As we may think," Atlantic Monthly, (July 1945): 101-108.

Christensen, Larry C. & Bodey, Michael. "A Structure for creating Quality Courseware", Collegiate Microcomputer, (Autumn,1990):201-208.

Christensen, Larry C. & Stokes, Gordon, Hays, Bill. Design Strategies for a Computer-Based Instruction System. ACM SIGCSE Bulletin, Vol.18, Number 1, February 1986.

Frasson, Claude, Gauthier, Gilles. "Intelligent Tutoring Systems-At the Crossroads of Artificial Intelligence and Education." Norwood, NJ: Ablex Publishing Corp. (1990).

Halasz, F.G. "Reflection on NoteCards: Seven issues for the next generation of Hypermedia Systems", Communications of The ACM 31, 7 (July 1988): 836-852.

Jonassen, David H. "Hypertext/Hypermedia", Inglewood Cliffs, NJ: Educational Technology Publications, Inc. (1989).

Nielsen, Jakob. "Hypertext & Hypermedia", Boston: Academic Press, Inc., (1990).

Novak, Joseph D. & Gowin, D.Bob . "Learning How to Learn", New York, NY: Cambridge University Press (1984).

Romiszowski, Alexander J. "The Hypertext/Hypermedia Solution- But What Exactly is the Problem?" Designing Hypermedia for Learning. Ed. David Jonassen and Heinz Mandl. New York: Springer-Verlag, (1990). 321-354.

Yankelovich, N., Haan, B.j., Meyrowitz, N.K., and Drucker, S.M., "Intermedia: The concept and construction of a seamless information environment," IEEE Computer 21,1 (January 1988): 81-96.

Appendix A ERDA Lesson Titles and Numbers

Module

Number	Title
1	Introduction, Review of Data Types, Arrays and Records
2	Data Types
3	Data Structures
4	Data Abstraction
5	Arrays and Records
6	Pointers and Simple Linked Lists
7	Stacks
8	Queues
9	Lists and Ordered Lists
10	Recursion
11	Trees-an Overview
12	Binary and Binary Search Trees
13	AVL Trees
14	Heaps and B-Trees
15	Sorting
16	Searches
17	Sets and Hashing
18	Character Code
19	Strings
20	Graphs and Graphs Algorithms

Appendix B Algorithm for Correction Strategy

```
PRINT task to test concept#1
READ solution
INITIALIZE LastMastered
WHILE (incorrect solution)
BEGIN
  FOR each link attached to concept#1
    PRINT link question
  READ student's choice of link
  PUT path containing link on OPEN LIST
  IF ConceptNum = LastMastered THEN
    ASSIGN NotMastered to concept
  WHILE (concept not= Mastered) AND (Open
       list not empty)
  BEGIN
    PUT path on Closed list
    FOR each link attached to next concept
      IF path is not on Open or Closed lists THEN
        PUT path on Open list
    IF ConceptNum = LastMastered THEN
      ASSIGN NotMastered to concept
END
IF concept = Mastered THEN
BEGIN
    FOR each link in path PRINT corectional text
  ASSIGN ConceptNum to LastMastered
END
ELSE
BEGIN
FOR each link attached to concept#1
IF path not=student's choice
THEN
PUT path on Open list
IF concept = LastMastered THEN
ASSIGN NotMastered to concept
WHILE (concept not= Mastered)
BEGIN
  PUT path on Closed list
  FOR each link attache to next concept
    IF path is not on Open or Closed lists THEN
      PUT path on Open list
IF concept = LastMastered THEN
      ASSIGN NotMastered to concept
      END
  ASSIGN ConceptNum to LastMastered
    FOR each link in path
        PRINT correctional text
PRINT correctional text of student's choice
          of link
    END
  PRINT task to test concept#1
    READ solution
END
```

Adopting Client/Server Computing: A Model for Small Colleges

CDR Steven M. Conway
Mr. Barry M. Levine
Department of Information Services
United States Coast Guard Academy
15 Mohegan Avenue
New London CT 06320
203–444–8349
conway@dcseq.uscga.edu

Abstract

The significant benefits of client/server architecture over traditional information systems are reduced server hardware investment and improved user interfaces. We are developing our client/server academic administrative system in-house in part because commercial packages for academic administrative systems don't currently support client/server architecture. While developing our applications we are attempting to re-engineer our business processes. We will discuss our system development approach to replacing our existing academic administrative systems with a client/server system.

Paper

Like many organizations, the U.S. Coast Guard Academy is adapting to an environment of decreasing resources. Administrative information systems that support greater demands with fewer resources often has been automation's clarion call. Usually, systems support greater demand at the same or greater cost. We are changing our administrative systems from a proprietary Unisys mainframe to UNIX-based Oracle client/server system partly to support greater functionality at lower cost. Rather than simply transfer the old hierarchical system with its old code to the new system, we are make dramatic changes. The development method, enabling technologies and strategies that make this possible include: Rapid Application Development (RAD), client/server technology and appropriate success metrics.

Key to our process were the following premises:

Rapid Application Development (RAD) at the U.S. Coast Guard Academy is based on two maxims: (1) Customers do not know what they need, and (2) developers do not know better than the customer what the customer needs. The consequence of these maxims is: the customer and the developer can discover what the customer needs by creating and using incrementally enhanced software in a compressed timeframe. The emerging literature refers to this as "spiral development". Although customers cannot say in advance what theyneed, they can use software and identify what it does incorrectly or what functions it doesn't support. "The miracle occurs when they touch the keyboard."

Traditional methods obeyed these rules too. Rarely has software delivered to meet written specifications actually met the customers' needs. Instead, software was incrementally modified during a phase typically called "maintenance" in a process we call "Slow Application Development".

Customer Focus Groups direct the U.S. Coast Guard Academy's RAD efforts. Each software project has a chartered Focus Group whose membership includes representatives of constituencies affected by the software. Each group: defines and redefines the desired software product features, prioritizes feature development, uses early software to do useful work, and clarifies business rules. The groups make decisions by developing consensus. Each group has a group leader who has authority to articulate Academy business rules. Information Services personnel serve on each group as advisors. Our customers own the process. Software development changes from an adversarial to a collaborative process.

Because the final product is achieved incrementally, one risks falling victim to "Scope Creep". This is the process of incrementally adding features in an environment of seemly limitless marginal resources. To address this issue, each group has a budget of developer hours. Within this constraint the groups prioritize development. We start each project's work knowing that the software will not satisfy all of our hearts' desires ? just those we can afford. We finish development when the money runs out, the same standard that applies to traditionally developed software. Developers and their customers accept this reality up-front.

Traditional methods all have a step entitled "implementation and acceptance". Because the people who need to accept the software use it during development, are party to its improvements and have approved all changes, acceptance is moot. Our implementation metric is simple: "Our customers will want to rip the software out of our hands." The corollary is that we do not force software on our customers. If the customer thinks that the products don't add value, we won't field them.

Additionally, we only build intuitive systems that require no user manuals. Commercially developed software is already migrating towards this standard. Traditionally, user manuals were done late, poorly and were rarely maintained or used. So we eliminate them. We modify our software until its functions are clear to all who may use the software. We provide on-line help where appropriate, and cueing with pull-down lists and menus.

We are attempting to re-engineer our business processes. For example, our Academic Planning client software supports the work of students and advisors versus supporting the registrar, a middleman, like our legacy system. We strive to make software that supportsthe work people do, rather that software that makes one person a data entry clerk for another.

Issues Faced:

Embarking on this project has challenged us. We have faced many issues, some were foreseen and some complete surprises. None were showstoppers.

CULTURAL - Our staff is well versed in traditional methods using COBOL and a hierarchical database. Development using RAD, a relational database and object oriented/visual programming based client/server development tools (Hypercard initially) required a major cultural change by our analysts. We acknowledged this as a major issue and addressed it before and during the process. Yet, making this cultural change continues after a full year. Our analysts willingly try new tools and techniques, but the depth of their development biases should not be underestimated. We have turned their world sideways; it takes time and effort to equilibrate.

BUY VS. BUILD - Before embarking on a large development project (with a small crew), we carefully evaluated the option of purchasing commercial off-the-shelf (COTS) software for our administrative systems. For us, about half of the COTS functionality would have fit. As a federal military academy, there are significantly different assumptions.

Our students do not pay to attend the Academy; the federal government pays them. This makes financial aid and bursar functions unusable. Also, we require systems that monitor and maintain data about military performance, demerits, physical fitness and regimental organization, and integrates them with standard academic data. Although some systems allow on-site customization, we determined that the cost of purchasing and customizing approximated the cost of just developing. The long-term costs to synchronize the COTS and customization tipped our decision to building our own software.

RESOURCES - We are funding our project by reallocating legacy system maintenance resources. Our legacy system customers agreed to make no software changes for two years, except those needed due to program or data error, or externally mandated business changes. Seventy-five percent of the analyst staff effort was redirected from the legacy system to new development.

CLIENT DEVELOPMENT SOFTWARE - Client/server development tools standards are nonexistent. Until recently one could pick a market leading development tool as well with your eyes open as closed. Cognizant of its limitations, we have used Apple's Hypercard for our initial development. We are looking for more robust tools that meet our requirements. To be considered, the tool must be deployable across different platforms (Macintosh and Windows at aminimum). The tool must be a visual programming tool with an object metaphor. Also, tools with run-time licensing requirements are excluded. Candidate tools include FoxPro Professional, PowerBuilder and Novell AppWare.

DUAL USE OF DATA WAREHOUSE TO SUPPORT PROCESSING AND AD HOC QUERIES - Our legacy system supports data processing. Ad hoc querying is onerous and accessible only to the extremely technically adept. The power of relational data bases is that they can support processing and ad hoc access. As we move functions from our legacy system to the client/server system, we transfer data and make it available for ad hoc query. We use COTS for ourselves and our customers. Currently we are using Brio Technology's DataPrism. By making data available early in the process, we greatly increase the value of our project at minimal cost.

MANAGEMENT SUPPORT - Re-engineering gurus all chant the mantra "senior management must drive the re-engineering train". We do not enjoy that luxury. We are succeeding with senior management's benign indifference.

Conclusions

Although our work is unfinished, we draw these interim conclusions. Small colleges can adopt client/server computing. We've had no show-stoppers. The most challenging aspects relate to people rather than technology. Specifically, changing a development group's culture required more time and effort than we expected. Collaborating with our customers has been extremely beneficial. Only practical experience convinced our staff that customers will gladly tolerate buggy beta software as the price for making incremental decisions about software products. Empowering customers and trusting their judgement isn't just a theory; it really works.

Background

The United States Coast Guard Academy (USCGA) is a federally-funded four-year college similar to other federal military academies. Approximately one-half of the U.S. Coast Guard officer corps received their B.S. degree and Ensign's commission at a USCGA graduation. The Academy has over 900 students, called Cadets, and confers degrees in eight majors.

The Academy requires each entering freshman (swab) to purchase a Macintosh computer. Most faculty have Macintosh computers. This broad base of Macintosh computers along with a fiber optic network developed in 1990-1993 provide the infrastructure that supports our client/server development.

The USCGA's legacy information system runs on a hierarchical data base accessed by COBOL programs hosted on a Unisys A6/FX mainframe. The system has the typical limitations of near nonexistent ad hoc access and extremely high hardware and software maintenance costs. Because the system is proprietary, almost any additions or enhancements were cost prohibitive.

The USCGA's client/server system is based on a remote database model. The database is accessed through an Oracle RDBMS and is hosted on a Sequent Symmetry S27 computer.

CDR Conway heads the USCGA's Information Service Department.

Mr. Levine is responsible for the USCGA's Administrative Systems.

Learning Efficacy and Student Reaction to Computer Assisted Instruction

Amit Dave, Ed Billmire
Department of Computer Services
Life College
1269 Barclay Circle
Marietta GA 30060
404-424-0554

ACKNOWLEDGEMENT

The author wishes to acknowledge the assistance provided by Dr. Ganga Persaud of the Dekalb County School System's Department of Research and Evaluation. His kind guidance and advice were an invaluable resource in the completion of this study.

Of course, the cooperation and interest of students involved in this study is also very much appreciated.

Objectives

1) To compare (a) the mean on-going test scores and (b) the final test scores of the experimental group with those of the control group;

2) To show the relationship of all demographic variables in the test scores and final scores;

3) To use demographic variables to account for variations in test scores.

Theoretical Perspective

Computer Assisted Instruction (CAI) has been successfully implemented in the academic curricula of several colleges and universities. In this study, the value of the modern CAI approach, as compared to the traditional method of education, is explored at Life College, using the subject of Biology for a pilot project.

The students' perception of CAI is believed to play a major role in its effectiveness. This theory is tested using the results of a survey questionnaire to obtain a measure of students' perception, and their test scores as a measure of CAI's effectiveness.

Based upon the results of this study, the feasibility of implementing CAI in areas other than Biology will be explored. The long term intention is to use CAI as a tool to enhance the total educational experience of Life's Chiropractic students, as theyprogress through various areas of study.

Assumptions

It was assumed that the majority of students in the study did not have any previous experience with the Computer Assisted Instruction method in the classroom environment. With CAI in the area of Biology, the performance of students would improve, as it provides graphic displays of human anatomy, and the visual images of botanical features would provide the student with a fascinating environment, making learning more interesting and enjoyable. Further, the students will post higher mean class grades compared to the traditional teaching method. Since the teacher initiates the use of CAI, and serves as the source of both direction and motivation, his/her attitude towards computer use directly impacts the students' perception towards CAI, and hence, on their class grades. Further, the computer enables a student to find his/her error on an ongoing basis, and hence, to be able to make corrections by himself/ herself or if frustrated to seek assistance from instructor to resolve problems. The CAI has the advantage of on–the–spot corrective mechanism as compared the long wait time of traditional classroom. Finally, a basic assumption was the lack of computer expertise on the part of all students.

Methodology

Students were assigned tasks from specific chapters and were asked to complete the task using Computer Assisted Instruction. Macintosh based software was used for this purpose. Lectures were facilitated with CAI, and periodic tests were given which were used in the statistics to derive the inference.

Towards the end the quarter, a questionnaire was administered, covering areas such as students' preferences about CAI use, and the application of CAI in other areas such as Physics and Chemistry.

The study was conducted using a fairly large sample size. The experimental group consisted of 74 students and the control group consisted of 61 students. The survey questionnaire contained 14 questions. Responses were ranged on a scale of 1 to 5, with 5 being the highest and 1 the lowest.

All data gathered from the questionnaire, test scores, and final scores were analyzed using SPSS statistical software.

Results

The results generated by SPSS are shown in Tables 1–4, and are discussed as follows:

1. In a comparison of mean test scores between experimental and control groups, the results indicate that
(a) The mean of the on–going test scores of the experimental group (885.98) is higher than that of the control group (794.42) as shown in Table 1. The difference is significant at .05 level;

(b) The mean final score of the control group (779.34) is higher than in the experimental group (624.79) as shown in Table 2. The difference is significant at .05 level.

Hence, CAI was better for the experimental group with respect to on–going tasks and tests but in the final examination, the control group performed significantly higher. It is possible that the control group started out with higher test scores. No record was kept of initial ability scores for the two groups.

2) In an examination of test scores (on–going and final) in relation to the demographic variables, the results of factor analysis (varimax rotation) indicate that:

(a) In Factor I, students with higher final scores recommended the implementation of CAI in other areas. They have positive comments, and their understanding is better in the area in which CAI is implemented;

(b) In Factor II,there is a positive relationship between the preference of using CAI and feeling comfortable with it, and the students' opinion that it is an effective tool for learning;

(c) In factor III, the more problems the student faced in completing computer exercises, the better his/her test scores. This could be attributed to the CAI enabling students to find errors, and to solve them with aid of computer, or if they felt frustrated, seek advise of the instructor;

(d) In factor IV, the majority of students exhibited a favorable attitude towards integrating CAI into the academic curriculum;

(e) In factor V, experience and improvement are inversely related.

The results indicate that ongoing test scores were improved by CAI probably because the computer program helped students to find and correct errors on an ongoing basis. However, the final scores were related to students opinions that computer should be included in other courses, that they understood concepts better, and had positive comments, etc. Final scores were, therefore, associated with higher order learning and satisfaction.

3) In a regression analysis of the data to explain what variables explain the on–going test scores, the results in Table 4, indicate that student problems (beta = .292), and student feelings of improvement (beta= –.290) make significant contributions to test scores. The overall variance, however, is over 5% as indicated by the adjusted R square.

The results support the relationship between problems and test scores in the factor analysis. In addition, students feelings of improvement make some contributions, though the relationship is inverse. Therefore, students seem to feel greater sense of improvement than the gains they make in actual test scores.

4) In a regression analysis of the data to determine what variables explain the final test scores, the results in Table 5, indicate that student who recommend use of CAI in other areas (–.293) make significant contributions to change in final test scores. The total adjusted variance is just above 5%. The relationship is inverse indicating the more they recommended the course, the less they performed on the final examination, and vice versa. The other variables did not make a significant contribution. This means that the main variables which explain variance in final scores have not been included in this study. Further studies should examine the relationships between initial ability and final scores.

Conclusions

Overall, CAI has been demonstrated to be an effective teaching and learning tool on an on–going basis, exercise by exercise, and holds great potential to play a major role in the modern classroom environment for this purpose. Long term gains by students are explained by other variables not included in this study.

It is possible when considering the effectiveness of any teaching tool, however, that the "good" student is able to reap the benefits of all his resources. The final scores might be explained by this argument.

It would appear from the short term gains made by students on the on–going tests, having resolved computer identified problems that the value of CAI may best be judged on the differences in performance of deficient students in CAI versus traditional classrooms. Thus further experiment is required to determine the performance of low achievers with CAI and traditional methods.

Table 1

Variable	Sample Size	Mean	Standard Deviation
Test Score			
Control Group	61	794.4262	99.424
Experimental Group	74	885.9865	97.547

Table 2

Variable	Sample Size	Mean	Standard Deviation
Final Score			
Control Group	61	779.3443	95.077
Experimental Group	74	624.7973	230.625

Table 3
Factor Analysis

	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5
Final Score	0.73132	-0.07814	-0.10228	0.18529	0.19720
Included	0.72433	0.23113	0.09125	-0.1601	0.00649
Recommendation	0.64501	0.29952	0.15819	-0.00156	0.07276
Understanding	0.56863	0.48291	0.06847	0.04719	0.01643
Comment	0.49123	0.46055	0.34383	-0.03777	-0.01541
Areas	0.40763	0.20602	0.38623	-0.11347	0.12906
Preference	0.13436	0.81346	0.13764	-0.04596	0.16579
Helpful	0.16422	0.75570	0.08368	0.01243	0.09468
Comfortable	0.12248	0.70808	-0.02948	-0.27722	-0.23339
Effectiveness	0.40980	0.47326	0.35337	-0.05534	-0.10985
Time	-0.14292	0.08870	0.80773	0.19823	-0.17850
More Exercises	0.39392	0.04298	0.72805	-0.07042	0.10389
Test Score	0.19294	-0.09258	-0.07381	0.86832	-0.07955
Problems	-0.31064	-0.08952	0.18661	0.70913	0.33444
Experience	-0.19121	0.03988	0.15333	-0.16182	-0.83694
Improvement	0.07011	0.26802	0.54374	-0.16923	0.62895

Table 4

Regression Analysis with Test Score as Dependent Variable and all other variables as independent variables

Multiple R	0.40405
R Square	0.16326
Adjusted R Square	0.05779

Variables	Beta	Sig T
Problems	0.292198	0.0055
Areas	-0.071173	0.4976
Time	0.018975	0.8631
Understanding	0.169859	0.152
Experience	-0.076951	0.4597
Comfortable	-0.039511	0.737
Recommendations	0.150296	0.2229
Improvement	-0.290169	0.0117
Effectiveness	-0.115578	0.3357
Helpful	0.005302	0.9638
Included	-0.045949	0.6943
More Exercises	0.168137	0.1788
Preference	-0.014854	0.909
Comments	0.03833	0.7775

Beta Coefficient indicates that for every unit change in independent variable, there is an amount of change indicated by beta coefficient on dependent variable

Table 5

Regression Analysis with Final Score as dependent variable and all other variables as independent variables

Multiple R	0.40405
R Square	0.16326
Adjusted R Square	0.05779

Variable	Beta	Sig T
Problems	0.100511	0.3625
Areas	-0.293754	0.0033
Time	-0.045088	0.6639
Understanding	-0.044855	0.6859
Experience	0.166534	0.0939
Comfort	0.116249	0.2917
Recommendation	0.190419	0.1022
Improvement	0.024355	0.8032
Effectiveness	-0.19561	0.0697
Helpful	0.126159	0.2654
Included	0.049188	0.6754
More Exercises	0.136128	0.2228
Preference	-0.178394	0.147
Comment	-0.128649	0.3154

There is an inverse relationship between experience and final Score. This means the less experience the student had with computer, the higher the final score and vice verse.

QUESTIONNAIRE

1. How comfortable do you feel using a computer?
(A) Very comfortable
(B) Somewhat comfortable
(C) Not too comfortable
(D) Not comfortable at all
(E) NO opinion

2. Have you ever used Computer-Assisted Instruction (CAI) at any time before this?
YES NO If YES, please explain.

3. What did you like most in the computer exercises? (You may pick more than one)
(A) Graphic displays
(B) Explanation of terms
(C) Drills
(D) Everything in the exercises
(E) Did not like anything

4. Do you feel the computer drills helped you to learn?
YES NO Please explain.

5. How much time did you spend on each assignment?
(A) 7 or more hours per week
(B) 3-6 hours per week
(C) 1-2 hours per week
(D) Less than 1 hour per week
(E) As little as possible

6. Would you prefer to do more computer exercises?
YES NO Please explain.

7. Did you encounter problems while working on the exercises?
(A) Often felt frustrated
(B) Did not receive help during lab when I needed it
(C) Had difficulty understanding instructions
(D) Frequent problems with equipment
(E) Other - Please explain

8. Has the CAI course helped to develop or improve your computer skills?
YES NO Please explain.

9. Do you feel that using the computer to study has increased your understanding of course material?
YES NO Please explain.

10. Would you recommend CAI to other students?
YES NO Please explain.

11. In what other areas of study would you find CAI useful? (e.g. English, Math, Chemistry, Physics, etc.)

12. How would you rate Computer-Assisted Instruction (as compared to traditional classroom teaching)?
(A) Very effective
(B) Somewhat effective
(C) About the same
(D) Not too effective
(E) NO opinion

13. Do you feel CAI should be included as a regular part of the academic curriculum?
YES NO Please explain.

14. Additional Comments: (Concerns - Criticisms - Complaints - Questions - Recommendations - Compliments)

The Impact of a Networked, Multidisciplinary-use Computer Classroom on the Whole Campus of a Small Liberal Arts College

Kathy Decker
Clarke College
1550 Clarke Drive
Dubuque IA 52001
319-588-6391
kdecker@keller.clarke.edu

Abstract

Clarke College established a networked, multimedia computer learning environment (classroom and authoring facility) that is used by five departments. The availability of the ELECT (Effective Learning Environment using Computer Technology) tools and resources has reinvigorated faculty, changed the attitude of the college administration toward increasing the support for technology use on the campus, created an example to use in other grant applications, and is proving to be an excellent recruiting tool. Faculty have used the ELECT facilities to create very investigative and collaborative situations for students that are appropriate learning experiences for today. Faculty and staff, as well as the institution, have gained leadership roles in the community because of experience and growing expertise in making state-of-the-art technology accessible to the college community and in enhancing curricula by integrating technology. The success of the ELECT project has fostered an increase in grant proposals that seek to expand the facility and extend the uses of the facility.

Background

Clarke College, founded 151 years ago, is a four year liberal arts college with 975 students. There are 19 academic departments with 52 full time faculty. The student population consists of 66% traditional age and 34% adult students. Clarke has a long history of using computers. The Computer Science department will be 30 years old in 1995. It was established by Sr. Kenneth Keller who was also very involved in the early years of the ASCUE organization known originally as CUETUG.

Elect Classroom/Lab

In the summer of 1992, Clarke College installed the ELECT (Effective Learning Environment using Computer Technology) Classroom/Laboratory. It was funded by grants from NSF/ILI and IBM with matching funds from Clarke College. The facility has three components: 1) a computer-based multimedia classroom with 10 student workstations, a multimedia control workstation, a multiscan color video/RGB projector with sound, a laserdisc player, a color scanner and printers; 2) an adjacent faculty authoring room with a multimedia workstation, a PC-VCR and monitor and network server; and 3) a mobile computer station identical to the control workstation with a color LCD projection pad and projector, VCR and large monitor/receiver.

Student workstations are configured with 486SLC 40MHz processors, 80MB disk, 8MB RAM, and VGA color monitors. The control, authoring, and mobile stations are configured with 486SLC 40MHz processors, 160MB disk, 8MB RAM, XGA color monitor, CD-ROM drive, optical disk drive, and sound board. The network file server consists of a 486 processor, 800MB disk and 10MB RAM. All equipment runs over a 16 MB Token Ring LAN and is networked using Novell Netware 3.11, DOS 5.0 and Microsoft Windows 3.1 with Multimedia Extensions.

The network provides a mechanism to prepare and distribute exercises authored by faculty. These materials can be accessed by both faculty and students during classtime or while working independently. Windows 3.1. Graphical User Interface (GUI), rather than a menuing system, is used as the end-user interface to the network to allow for multi-tasking at the workstation level. A single exercise stored as a file can be accessed by every workstation via the network. Through the network, students have the ability to share printers and the output from the scanner, laser disk, CD-ROM and optical drive. Student assignments submitted to the faculty can be retrieved, evaluated and returned via the network. With LANSCHOOL software, faculty are able to direct screen information to any or all student workstations and students' work can be monitored. LANSPOOL software is used to direct printing. The ELECT network has access to the campus Internet host.

Learning Experiences

Faculty use the ELECT facilities to create investigative and collaborative situations for students that are appropriate learning experiences for today. Initially the ELECT project focussed on freshman-level courses in Mathematics, Biology, and Computer Science. Following national trends, Clarke students interested in math and science as freshmen had sometimes lost interest in a major in the sciences by the end of their sophomore year.

In the first semester of the ELECT classroom use in the fall of 1992, seven courses involving a total of seventy-five student users were taught by six faculty from the Biology, Computer Science, and Mathematics departments. In the Fall 1994 semester, at least eleven courses will be taught by nine faculty from the original three departments and Communications and Business. Outside of scheduled class periods, the ELECT classroom is available to students as an open lab.

Some faculty involved in the ELECT project initially had very little experience. The classroom tools were quickly learned and these faculty are now a resource to training others in the institution and community.

Faculty Reinvigorated

The ELECT facility, with its new tools and resources, has reinvigorated faculty. By using this new facility, they are able to conduct hands-on classes, incorporate multimedia techniques, utilize software to demonstrate powerful learning examples, and use tools to allow students to freely explore ideas. Faculty are excited by this new robust way of teaching.

Mathematics Department faculty with many years of teaching experience in the traditional classroom setting have dramatically changed their teaching style through the incorporation of hands- on experiences with Mathematica. To initially become familiar with this software package, they attended summer workshops funded by NSF and The PEW Foundation. The powerful commands of Mathematica allow them to use more complex examples than previously possible. Students are no longer limited by what they can do with only a calculator. They can now graph equations with the computer indicating the limits. Both faculty and students can tinker with equations letting the computer demonstrate new concepts, thus fostering a better understanding of mathematics.

Biology has incorporated a closed lab session to provide students with the analytical tools and accessibility to resource information for the investigative interpretation of wet lab results. This has lead to a greater understanding of the concepts behind the lab exercises. Students continue to use these tools formally and informally in studying and reporting lab results beyond the first Biology course. With the ELECT laboratory tools, Biology faculty can now use new approaches in their teaching. New tools available through the Internet can be used to focus student research. Additionally the increased communication between faculty and students through electronic mail has enhanced the quality of the students' work.

With the availability of this ELECT classroom, Computer Science has been able to add a one-hour closed laboratory to its introductory course. Faculty find that the interactive nature of the networked control station and student workstations combined with the presentation capabilities in this computer classroom provide the opportunity for immediate feedback on a student's work. Students are able to access class notes, assignments, and data files through the network.

The state-of-the-art ELECT technology allows us to meet the demands of faculty in need of multimedia training. This summer Clarke College will host a workshop which addresses the design and implementation of interactive classroom presentations and student lab assignments, screen design and interface building, scripting, integration of music, voice, video, graphics, and animation. Faculty from 14 departments will attend this hands on workshop using Asymetrix Toolbook, Express Author software, and action media. Concepts learned in this workshop will be applicable on both MACINTOSH and IBM platforms. The excitement generated by this workshop and the production of multimedia applications by faculty incorporating discipline-specific materials will enable technology to take another big step on the Clarke campus.

Additionally, the ELECT facility gives the campus easy access to the Internet. This avenue to the Internet, with its unlimited access to databases and library resources, has allowed faculty to incorporate new teaching methods into their coursework. They are able to stay up to date with current developments in their disciplines through newsgroups and communications with colleagues from around the world.

College Administration

The ELECT classroom/laboratory has increased the awareness of the college administration of the growing need for technology across the campus. The administration has seen faculty reinvigorated and the effect that technology can have in stimulating learning. The ELECT classroom is also used to train administration on PC applications and the Internet through workshops and classes offered by the faculty.

Administrators are now familiar with the use of multimedia, the Internet, and campus networking as enhancements to teaching and learning. They are aware of the effects of the ELECT facility on the campus community and are thus able to build upon this success while planning for future technology and seeking grants for its funding. A list of desired technology applications has been upgraded to a master campus technology plan, which now serves as a goal statement when applying for grants and approaching funding sources. In addition, this master plan can be used to provide the "big picture" to faculty, staff, and students during planning sessions.

Since the installation of this facility, increased support for technology was met by hiring a network and microcomputer coordinator and installing an IBM RS/6000 to enhance academic computing and serve as the Internet host.

Grant Implications

The success of the ELECT facility has served as a stimulus for the entire Clarke campus toward continuing technology development. The ELECT classroom layout, equipment configurations, and classroom scheduling are used as models for additional grants. A Title III grant has requested funding for a similar configuration of a MACINTOSH multimedia classroom, faculty authoring, and mobile environment. Departments involved with this MACINTOSH platform have received Title III funding for faculty development this summer. In addition, Clarke has been chosen as a regional training site for an Internet workshop funded by The Annenberg Foundation. A FIPSE grant that would have involved working cooperatively with a high school science program was not funded. We are awaiting word on an Apple grant involving the Clarke Education Department and a rural high school in incorporating technology in high school coursework. Ongoing meetings of the ELECT group explore other funding possibilities that seek to expand the ELECT facility and extend its use.

Recruiting Tool

The ELECT facility serves as a showcase to prospective students. They are able to see the investment that the college has made in fostering state-of-the-art teaching and learning and are excited in the prospect of its use. Interactive, hands-on learning also helps in retention of our current students.

Local businesses have also used the ELECT facility as a model for their own networking or multimedia applications. This has resulted in increased course enrollment because of their employees' need for additional training.

Leadership Roles

Faculty and staff, as well as the institution, have gained leadership roles in the community because of experience and growing expertise in making state-of-the-art technology accessible to the college community and in enhancing curricula by integrating technology. Faculty are working closely with textbook publishers in developing materials and reviewing new textbooks. Faculty members also serve as guest speakers at professional meetings, brown bag luncheons and specialized workshops directly related to the capabilities of the ELECT facility. Computer Center staff are an increased source of information for many companies on network and wiring design, implementation, and support.

Advice For Success

For other institutions to achieve the kind of accomplish- ments outlined above, it is imperative that a core group of interdisciplinary faculty and staff take a leadership role on campus. This group must investigate interdepartmental needs, find funding sources, and write grant proposals. They should also spearhead follow-up activities such as faculty training and curriculum development.

Conclusion

The ELECT facility has been the impetus for many changes on the Clarke College campus since its installation two years ago. This is not the end of our story, but the beginning of future technology developments on campus. Administrators and faculty are aware that continued and increased support and enhancements are critical to a vibrant future for Clarke College.

Growing Pains!!!! Where Do We Grow From Here???? Planning the Growth of Computing Resources at Rhode Island College

Allyson B. Dickie
Rhode Island College
Providence RI 02908
401-456-9806
dickie@ric.edu

Computing at Rhode Island College

The Last Five Years - 1989 To 1994

Between 1989 and 1994 computing activities at Rhode Island College changed dramatically as they have everywhere. 1989 - 1990 was a quiet time. 1991 - 1993 was a steady growth period with the introduction of new hardware and software. Now we see an exploding demand for more sophisticated resources in traditional classrooms, computing labs, and offices of administrative end users.

Due to rapidly changing technology the future direction of computing is NOT CLEAR, and traditional sources of funds for budgeting future growth are "drying up". These factors present enormous challenges for planning and funding future growth.

Below are growing pains that we are experiencing, strategies we are using to accomodate this exploding but unclear demand in the face of shrinking budgets, and alternatives we are considering to determine where we might grow from here.

Where Were We?

1989 - 1990

Five years ago, computing activity was neatly defined and entertained a relatively small audience. There were two separate computing activities -- academic and administrative computing. User training consisted of leader led workshops, written tutorials, and individual consulting. Since Rhode Island College is a public state institution, our funding came from tuition and state appropriations.

ADMINISTRATIVE COMPUTING. Eight programmers serviced most data processing needs for administrative offices by writing batch programs and online CICS programs to prepare summary reports and for transactions processing against the administrative data base on an IBM 4381 mainframe. Some offices used DOS PC programs for wordprocessing and spreadsheet functions. Our Admissions Office downloaded data for accepted students to an Alpha III data base which was later integrated with WordPerfect to process acceptance letters. One full-time consultant supported this PC activity.

ACADEMIC COMPUTING. The Office of User Services operated six labs and employed a full-time consultant and twenty-five students to support academic computing.

IBM 4381 mainframe/VAX 11/780 minicomputers. Two dumb terminal rooms -- one of VT100 terminals and one of VT240 color graphics terminals -- gave access via our campus backbone ISN fiber optic network to our IBM 4381 mainframe and our two VAX 11/780 minicomputers. Two programming courses, taught in the VT100 room, used the mainframe. We also used the mainframe to access BITNET, the wide area network for communication among institutions worldwide. BITNET was not easy to use, and it did not catch on. Other programming courses and courses doing statistical analysis used the VT100 room to access one of our VAX 11/780 minicomputers. A history program of computer aided instruction accessed the other VAX from the VT240 room.

Apple/Macintosh personal computers. Use of Apple/Macintosh personal computers and software began to expand in this period. Education courses used an Apple II lab, and a lab of Macintosh Plus computers. User Services did not manage these labs, but we did implement a $20,000 lab of six Macintosh IIsi computers to give students more time to complete projects since the other two labs were run by faculty and were available for only limited hours. These labs used a local talk network to share printers. Microsoft Word was used for wordprocessing, Aldus Pagemaker for desktop publishing, and Microsoft Works to provide database and spreadsheet capabilities.

IBM/DOS personal computers. Three DOS labs supported courses in computer literacy. Our Committee on Academic Computer Usage chose the following PC applications as the campus standard -- WordPerfect for wordprocessing, Lotus 1-2-3 for spreadsheet work and dBASE III+ for the database program. We used terminal emulation software for terminal activity on our VAX minicomputers (PC-VT) and IBM mainframe (Y-TERM), and we used Kermit for file transfer to/from both systems.

One lab had 8088 IBM PC's with a 30 megabyte internal hard drive and one 5 1/4" floppy diskette drive. They started from the internal hard drives. The Automenu program listed DOS applications supported by User Services. Users accessed the program they needed by activating the corresponding menu entry. Automenu is now the standard user interface. These computers had minimal capacity by today's standards. We were proud of them, however, and they serviced us better than adequately until 1992 when they were replaced and recycled to faculty who did not have a PC. Each of the other two labs contained stand alone 8088 IBM PC's with two floppy diskette drives which were started by means of a DOS startup diskette. Once DOS was active, users replaced the DOS diskette with a program diskette in order to run the program of their choice. These labs used switch boxes attached to one printer in each of four rows; four computers were attached to each printer, but simultaneous print activity was limited to four people at one time.

Our growing pains began in the end of the summer in 1990. With no experience in networks we replaced one of the labs with two floppy drive systems with a $70,000, token-ring local area network. Due to rapid advances in hardware, software, network, and internetwork technology we are still climbing the learning curve. We are, however, learning valuable lessons by experience, painful as they are at times.

The file/print server was an IBM PS/2 model 70 80386 computer with 16 megabytes of RAM and 110 megabytes of hard disk storage. The network software was OS/2 LanServer. The workstations, IBM PS/2 80286 PC's, had one megabyte of RAM and two 3 1/2" disk drives. Three had external 5 1/4" floppy drives so users could access files created on the old systems. They had remote startup capability so that a user could access the network by pressing the power switch. Users then had to "log in" to access programs and files on the fileserver. We used LanServer's menu system which was different from Automenu, and we loaded existing DOS programs on the fileserver without additional software cost. We established backup to the network with sets of startup diskettes for each application so faculty could conduct classes from diskette if the network "crashed".

Entry into the network world was initially an exhilarating and rewarding experience. The director of the Computer Center and I quickly learned the basics of defining print queues, defining users, and giving users the necessary privileges to access print services and applications. We learned most of this ourselves with some assistance from IBM systems engineers. Benefits of a network were immediately obvious. System startup and application access was much faster. Two as compared to four printers serviced the lab; many users could send print jobs at the same time while print queues handled the traffic. BEST OF ALL, there was a single point of maintenance on the file server for twenty workstations. We left the dark ages! Life was wonderful!!!

Our enthusiasm was shortlived, however. We quickly and painfully discovered the major liabilities of networks. Our system crashed in full laboratory conditions daily; the deadly single point of failure reared its ugly head. When the server crashed, every machine in the lab crashed. BUT... we were smart enough to have back-up diskettes? Yes, but faculty liked the benefits of networks and groused loudly about teaching from diskettes, the only method they knew just a few months earlier. I was one of those teachers, and it WAS painful to use diskettes again. When the lab crashed during class, the flow and valuable teaching time were lost. Since printing was also controlled by the fileserver there were NO backup print services when it crashed.

Given our inexperience with life in the network world, we were positive we had made mistakes setting network parameters. We never dreamed that the problem was with the LanServer software which was relatively new to the market. After suffering with this system for a full year and after much assistance from a competent and congenial IBM system engineer, we dumped LanServer in favor of Novell 3.11 at the cost of IBM. The lessons we learned from these growing pains guided us through our next growth phase.

Where Are We Now ?
1991 - 1993

TRANSITION

During the next three years, we witnessed continued growth in all computing areas. The major problem was the state fiscal situation. During 1989 - 1990 students paid one third of the cost of their education with tuition and fees, and by the end of 1993 they paid 50% showing the reduction in the state's ability to support academic activity. Getting additional personnel to support user demands for more support certainly did not look promising. How would we cope?

ACADEMIC COMPUTING. In 1991, we joined Internet, a bigger worldwide network offering more capability and easier to use than BITNET. By 1993, we dropped our connection to BITNET. From 1992 to 1993, Internet traffic increased by 233% from 750 to 2,500 thousand packets transmitted. The major users of Internet were faculty and students. User Services offered several new workshops and our individual consulting activity increased just for Internet use. We also processed VAX accounts as requested which increased the amount of clerical activity for account maintenance.

In 1991 we replaced two VAX 11/780's with a new VAX 6210 for more capacity to support Internet; this reduced our maintenance costs and gave us more floor space in our main computer lab. With some savvy negotiation by the Computer Center Director and Digital Corporation, we installed a Pathworks network of Digital 80286 PC's with two floppy drives -- one 5 1/4" and one 3 1/2" as part of the deal for the new VAX. We moved the PC's with hard drives to the lab that still had systems with two floppy drives which we again recycled to faculty. The result -- a new more powerful and less costly VAX, better equipment for that lab, an older lab upgraded, and faculty who did not have a PC got one all with minimal new cost.

In 1991, we also made the change from the LanServer network to Novell. I was the only full time person on staff, and because of our difficulties with LanServer, I kept the network simple – one print queue, one group of users -- purely vanilla. I also used Automenu here and on the Pathworks network so users could go to all three labs with minimal difference. In addition to ease of use, using one menu program meant that we only had to learn and support one program. Automenu also offered flexibility for allocating/releasing drive allocations in the network environment, and it was inexpensive compared to other menu systems. Applications also had the same default settings in each lab.

By 1992, operation of the two Macintosh labs which had been run by faculty was transferred to User Services in order to centralize support activity for software and hardware. This also resulted in an increased demand for services in an area where we had comparatively little experience.

In the summer of 1992 we added a fulltime consultant to the office without adding a new position and additional cost to the college. This person was transferred from another department where the demand for services was shrinking. With more labs to staff, and more software to support, the addition of one full time person was not enough. I had to increase the student staff from 25 to 35, by adding several higher paid lab supervisors and more monitors at an added cost of $40,000 -- nearly double what it had been.

In 1992, we also moved our "286" Novell lab to replace the stand alone PC's with hard drives; once again we distributed the older machines to faculty who still had need of a PC. We thought a "simple" move would be easy, but we used ethernet connections here instead of token ring. The network locked up every other day. Again, we thought our inexperience with ethernet wiring was the problem, but one semester later we discovered that putting an upgraded network card in the network server solved our problem. The problem was with the older card, not with us.

We installed a brand new Novell lab with an 80486DX2 file server and 80486SX workstations. Computer Information Systems faculty requested this equipment so that they could expand into the Windows arena for teaching. We had run Windows on the 286 network by upgrading the RAM on those PC's from 1 megabyte to 4 megabytes, and it is running there today; it runs quite nicely, but future needs indicated that we had to upgrade soon to keep pace with technological advances.

We also replaced the Apple II lab with Macintosh IIsi's in 1992 because the Apple computers were very nearly extinct animals. Even this time, however, we redistributed them to the elementary school which serves as a training lab for our Education programs. Macintosh use was still in its infancy and these IIsi's were a welcome step up from the Macintosh Plus PC's in the other lab (this would not be so two years down the road).

We had originally planned to purchase 80386 equipment and Macintosh LC's for these two labs. For once, however, the time delay required to process state requisitions turned into a benefit. By the time our original funding request was processed, technology had advanced and prices had fallen enough so that we were able to purchase the next higher level equipment. We were not, however, able to network the Macintosh lab other than for print purposes, and we were very quickly reminded of the nightmares of maintaining stand alone, machines equipped with individual hard drives. We have since installed At Ease on these machines which has reduced accidental trashing of applications but has presented other problems. The combined price tag for these upgrades was $90,000.

By 1993, the Math department wanted to use Maple under Windows. This required that we convert the VT100 terminal room to another Pathworks network of IBM 80486DX. The cost of this lab would be $70,000. To justify their request, the department submitted a detailed proposal citing research indicating the need for computing in the math curriculum and giving the numbers of classes and students to be affected. Given the reduction in state support, I doubt this request would have been funded without such substantiation.

Other academic disciplines also began using computers in the curriculum -- Biology, English, Art, Music, Communications/Theater, and Industrial Technology. There is also an increasing demand for support of the graphical user interfaces of Windows and OS/2 because faculty who do get new PC's are finding those systems already installed. Use of lab facilities for class instruction increased by 24% from the Spring, 1993 (864.5) total hours to the Spring, 1994 (1032.5) total hours, and some reservations could not be scheduled due to time conflicts. The increased use of labs for classes further stressed the availability of walk-in facilities for students to complete homework.

Faculty are also requesting large electronic auditoriums and classrooms, and there is an even greater demand for training and software maintenance from User Services. We were lucky again in January, 1993, to move a programmer from the administrative team to User Services. With the demand for new services now exploding, however, this addition has not even allowed us to keep pace. All three of us now are busier all day than I was when I was alone in 1989.

As mentioned earlier, I was forced to increase the size of my student staff. I was "fortunate?" to find one outstanding student in each of the Macintosh and DOS/WINDOWS areas. The downside to that fortune is that we have become heavily reliant on them to design/maintain our lab systems. We have been busy just "fighting fires". Recognizing this vulnerability, I have hired two new students one year in advance to pick up when the first two graduate. I am still uncomfortable about leaving this much expertise and responsibility in the hands of students with little full-time backup. Some attempts to remedy this problem are outlined in the next section.

ADMINISTRATIVE COMPUTING. PC use by administrative users has also increased in these years, and administrative users are also using Internet increasingly even if primarily for electronic mail communication on campus. One of the Computer Center's clerical staff is now assisting the fulltime consultant on a half-time basis. It appears at this time, that the distinct boundaries between administrative and academic computing are becoming blurry. More of the skills required to support both areas are the same, and with the anticipated use of networks in administrative offices, this will become even more the case.

By 1993, administrative use of the 4381 mainframe had outstripped its capacity. A new 9221 IBM mainframe was purchased at a cost of more than $300,000. The combined cost of academic and administrative upgrades that year totalled about $400,000.

Where Are We Growing????
1994 --->

One thing is clear. Growth of computing is continuing at an unprecedented rate. What is not clear is what direction this growth will take or how we will be able to continue funding it at a minimally anticipated yearly cost of $100,000 over the next five years.

The major planning goals that we identified during the last five years were to upgrade one lab each year, incorporate an electronic classroom in each building, and put a PC on every faculty desktop. We have succeeded in accomplishing the first of these, but we have not been able to accomplish the other two. The major obstacle to achieving our goal continues to be serious budget constraints. Labs with expected lives of five years are being funded with twenty-year state bonds.

We are, therefore, taking a hard look at our current structure to identify alternative means of maximizing services while minimizing costs and our dependence on state funding. To quote our President, John Nazarian, in remarks to the College on April 15, 1994, "To save the tree now, we must do some different kinds of pruning, bearing in mind that pruning is an opportunity for new growth." This section examines new alternatives for supporting technological advances.

First of all, we learned that repositioning current college staff is effective. Both people transferred from other areas had suffered some "burnout" and had developed skills that could be used more effectively in User Services. We are looking at other possibilities for repositioning and consolidating college staff. We have proposed that a member of the Audio-Visual department be transferred to User Services. He has Macintosh experience and experience in presentation media that we currently have no time to develop. With the expansion of computing into all academic disciplines, and the request for electronic auditoriums and classrooms, we would be able to implement a sorely needed faculty development area. Again we would be able to add new personnel to User Services by making use of another employee's skills which are no longer as useful to his current area; we could add new services without adding new costs.

We are also thinking about restructuring the computer center staff to make a separation between administrative programming and end user support rather than academic support. This would allow us to incorporate the consulting/clerical support staff in both directions. We would be able to provide better backup for the one administrative consultant for vacations, illness, etc. and she would be able to assist us as well.

We want to centralize "special" use equipment such as laser/color printers, color scanners, laser disc players, and CD-ROM drives. We are considering having a large walk-in lab at each end of the campus equipped with everything students need to complete homework assignments.

The other labs could be used for classroom use only. Scheduling activity, which has very little to do with our own work, could be given to the College's scheduling officer. We could then concentrate more of our efforts on our consulting activity. We could also reduce our student budget because we would no longer need to put monitors in these class labs, and with the resulting savings we could provide more capable student staff in the walk-in labs.

In the Fall, 1994 semester we will begin to issue permanent VAX accounts to all current faculty, staff, and students for Internet use. Right now student accounts are good only for a semester, creating additional paper work for account processing. By automating this effort, we will again be able to focus more of our attention on consulting activity.

We now have four separate local area networks. Our two Novell servers have licenses available that we have not used. We could add more workstations (including Macintoshes) to our proposed large walk-in labs without incurring additional network software costs.

We are investigating internetworking these separate areas. Before we hire a costly consultant to tell us what to do, we are requesting proposals from several competing vendors. Since the industry is so competitive right now, companies are willing to do this and we hope to be able to choose the lowest cost alternative.

Workstations would not have to be equipped individually with CD-ROM drives; they could tap into a server that uses a CD racking device. Neither would there have to be individual laser printers in each lab. Tying many local networks together and also giving them access to our VAX and IBM mainframe will be a costly undertaking, but it may save us a lot of money and aggravation in years to come.

We have one copy of every software application such as WordPerfect for Windows for every workstation. Reality in the walk-in labs is that not every user is using the same program at the same time. The same is true in administrative offices. We are testing a software usage tracking program called Softrak to get a better idea of how often each program is used and also how much use the lab is getting at different periods of time. We plan to use the results to determine exactly how many copies of each application we need and to determine our staffing needs; we hope to reduce expenses in both areas. Minimal software upgrades last year required in excess of $20,000. Hopefully, we can reduce this cost by moving to a license by usage approach as opposed to a license for every machine.

In the area of training we are investigating the use of video tapes, audio tapes, and online tutorials provided with software. We are also starting to use student-led workshops. All of these methods are less expensive than using full time staff to run workshops. Students work for a lower hourly wage than full-time staff. We purchased videos at a cost of $49 each; they can viewed many more times than a full time person could give an in-person demonstration for the same $49. Workshops led by full-time staff are also limited to the traditional Monday to Friday, 9-5 time frame; these other methods open training opportunities at virtually any time.

We are also starting to download written tutorials from Gopher sites that we can modify for our own purposes. This saves us valuable time from completely reinventing a wheel someone else already invented. We hope to distribute them electronically by means of our own Gopher soon in an effort to reduce paper and printing costs.

One of the more difficult problems we are facing is how to position ourselves to accommodate growth in new directions when faculty are not really sure of what they want to use today never mind tomorrow. Thank goodness -- the power PC technology is ready so both WINDOWS and Macintosh applications can run on the same PC.

Our next lab upgrade for the Macintosh plus lab will most likely be a purchase of Macintosh Power PC, hopefully this year because our IIsi lab is not able to support new software such as Photoshop and Macromind Director used by Art courses and courses in Communications/Theater. The really scary thing is thinking about what the next three to five years of technological advances will bring to the horizon that users do not know about yet and will most certainly want when it arrives.

Of course, the major problem is and always will be MONEY. It is evident that the state economic situation will not improve anytime soon, and computing needs will require more than what the state will provide. We must identify sources of funding other than monies gained from tuition and state appropriations.

First, we are considering charging special fees dedicated to technology acquisitions. The problem with this approach is that it is an additional cost burden to students not much different from a general rise in tuition. The benefit would be that those funds would be specifically targeted to technological upgrades.

There has also been discussion about charging fees for laser printing as a cost coverage mechanism. One problem is any fee system must be approved by our Board of Governors for Higher Education, a time-consuming process. The advantage would be twofold. Students would not be so likely to abuse laser printing privileges as they do with unlimited free printing, and costs for paper and expensive toner cartridges could be covered without impacting on the general College budget.

We are beginning to work with our Grants Administrator to seek funding from both public and private grant providers. We are hoping to take a college-wide approach to applying for grants so that we can get the best opportunities for the entire campus. This will involve a coordinated effort to identify a college-wide plan for technological needs, and grants are awarded more often to institutions with clearly defined goals.

In 1993, the College was awarded a grant from the Champlin Foundation which enabled the installation of a sorely needed multi-media language lab needed and a subsidiary multimedia Macintosh lab for multimedia development. The College is very proud of these facilities which are moving us in the direction we need to go. One problem, however, with separate departmental grant awards is that only those departments with those special needs are serviced. We hope that by bringing all departments together under a planning umbrella, we can get a "bigger bang for our grant bucks".

Summary and Recommendation.

In summary, the need for technological resources is growing rapidly, and traditional funds are shrinking just about as rapidly. This situation requires that small educational institutions be especially diligent about looking for new ways to support technological growth. The strategies that have kept us afloat at Rhode Island College and to date have allowed us to maintain steady growth are the following:

1. KISS -- Keep It Simple Stupid! Do not grow beyond your ability to support the technology you have in place. As we established each of our networks, we kept them simple using their basic functionality to start. We are slowly adding features such as different types of printers in the same print server and multiple user groups for special purposes. The key is to grow incrementally. Be comfortable with the fundamentals before playing with advanced features.

2. Standardize hardware and software whenever possible. This eases the troubleshooting process when problems arise. Having a standard user interface is easier to maintain and makes it easier for students to move from lab to lab. We have stayed with IBM/Digital DOS PC's for reliable service and technical support; having only a few types of machines also eases the burden on campus technicians.

3. Recycle equipment until it dies. -- Use networks to extend the life of older less powerful equipment, and to centralize special purpose equipment (CD ROM drives, scanners, printers, video cameras, etc.) in order to minimize staffing and hardware costs while increasing accessibility to users. Pass on replaced equipment to faculty who need it.

4. Stay flexible and tuned in to the market. You may be able to purchase more sophisticated equipment due to price reductions between the time you submit your initial plan and the time you purchase.
5. Research low-cost solutions. The industry is highly competitive due to rapidly evolving technology, and it is worth the time to find a solution that saves you money as we did when we purchased our new VAX.

6. Use phased-in growth. Only buy now what is necessary and what your user population is ready to use. Set up a faculty development area with state of the art equipment. By the time people are ready to use it, equipping labs and classrooms will undoubtedly cost less, and the technology will be improved.

7. Examine your personnel situation - student/full-time staff. It is likely that college-wide restructuring can centralize support functions. Better use of student staff allows full-time staff to be more effective.

Again, our President, John Nazarian points out, "We must, however, accept the reality of the fiscal crisis and the reality of change simultaneously. I am ready to work with each member of the college community not only to maintain the identity, quality, and integrity of the College but also to insure its growth in the future. Together we will nurture this tree -- my and your Rhode Island College...Nothing is easy and friends, what we have to do is not easy, but it is essential." The need for cooperative effort and new approaches in times like these is clear.

An Analysis of A Computer-Based Self Study Module: Student Reactions

William J. Gibbs
Media Services
Donald Armel
School of Technology
Eastern Illinois University
Charleston IL 61920
217-581-6011
cfwjg1@eiu.edu

Abstract

For Industrial Technology classes at Eastern Illinois University's School of Technology, computer-based learning modules are being developed using HyperCard. This approach was implemented primarily because it has the potential to, among other things, provide individualized instruction to students thus freeing up the instructor to spend more time on advanced concepts.

In the Spring semester of 1994, a pilot project was undertaken using two classes of junior/senior level Industrial Technology students to assess a computer-based self-study module. A survey was administered to collect subjects' reactions and the computer-based program incorporated tracking techniques in the software that provided insight to the students' use patterns.

This paper discusses reactions to the computer-based self-study method and presents the results of the data compiled from the attitudinal survey. Also discussed is a method used for tracking students' program use patterns.

Introduction

The development of computer-based instruction (CBI) is part of the professional education and instructional technology students' program curriculum. The CBI classroom provides a safe setting for practice and experimentation with concepts associated with self instruction using the computer. Putting those same CBI techniques into a real class, your class, carries a greater burden of risk.

A self-study module using HyperCard was developed to use in a Graphic Communications course to both improve instruction and take advantage of educational technologies. At the outset of development two main questions about the self-study module existed: how do we know it has improved instruction? and how do we know we have taken advantage of educational media? In either case it is easier for things to go wrong (educationally) than it is to meet learning objectives through accurate content design while using appropriate media. So the initial focus on development was benefit to students indicated by improved attitude toward the content, knowledge of the content, and motivation to learn.

Background

Computer-based instruction can be an effective method by which to attain instructional objectives (Gleason, 1981; Splittgerber, 1979), and when used appropriately it can be a powerful learning tool (Hannafin & Peck, 1989). The capacity of any medium to effect learning differences is contingent on how its capabilities correspond to the learning situation, the instructional task and individuals involved, and the extent to which the medium's potential is utilized by the instructional design (Kozma, 1991). Research on the effectiveness of CBI indicates that generally information retained from well designed CBI lessons is at least as good as that from more traditional modes of instruction (Dence, 1980).

Computers are especially useful in helping students build associations between symbolic domains, (e.g., graphs) and the real world phenomena they represent (Kozma, 1991). CBI specifically offers some distinct learning advantages, such as increased interaction, individualization, motivation (many students find it motivating) (Clement, 1981), immediate feedback, learner control, and easy of record keeping (Hannafin & Peck, 1989). A meta-analysis of CBI found that when CBI was used students had an improved attitude toward class work. Relan (1992) discusses some other benefits of CBI as being increased time on task, and non-judgemental feedback.

An important consideration in the design and development of CBI is the human/machine interface. The interface not only effects the delivery of instructional content but also how the software is used. This is especially true in tutorial programs. Factors which must be addressed are that tutorials be easy to use, their related concepts have a consistent interface design, and interactions required of students be similar (Orey & Nelson, 1991). It is from this perspective that in the Spring semester of 1994, a pilot project was undertaken to assess a computer-based self-study module. The user/machine interface was evaluated as was the program's instructional adequacy.

Purpose of Study

The researchers sought to investigate the potential of self-instructional CBI modules for Industrial Technology. A current problem with classes is that valuable instructor time is spent explaining rudimentary concepts, time which could otherwise be used for more difficult or advanced content. To help circumvent this problem and to supplement class lectures a computer-based self-study module was proposed for students' independent use.

A pilot study was conducted on two classes of 15 junior/senior level Industrial Technology students to investigate whether or not a self-study approach to instruction was viable. An important component of the study was evaluation of the module which is an extremely valuable aspect of any systems-based approach (Dick & Carey, 1985). The evaluation was designed to identify the benefits and pitfalls of CBI and explore a method of evaluation in which students' patterns of program use were tracked and used to refine the program. Lastly, the study looked at students' reactions to the module, the instructional approach and their perceptions as to how the module effected their learning. The study objectives are listed below:

- identify the benefits and pitfalls of a computer-based self-study module used to supplement class lectures;
- define a method for assessing a computer-based self-study module used to supplement class lectures;
- describe learner reactions toward the module and assess if it impacted their attitude toward learning.

Methodology

The sample population consisted of 30 undergraduate students who were either Industrial Technology or Graphic Design majors enrolled in two Graphic Communication courses at Eastern Illinois University. Subjects participated in the study as part of a class assignment.

The content for the self-study module was based on concepts pertaining to imposition, a graphic arts process in which pages are laid out in a press form so they will be in correct sequence when folded and bound. Based on this content, the researchers developed a computer-based module using HyperCard authoring software. It was run on the Apple Macintosh computer.

Procedure

Subjects received an initial demonstration of the software and its functions. They were instructed to sign up for a one hour session to use the module. Subjects used the software individually in a Macintosh computer lab under the researchers' supervision. While using the program the following data was recorded by the software:

- Student comments: Students could type comments into the software;
- Button clicks: The program featured a glossary, electronic note-taking and student comments facility. The program tracked how often students used the Glossary, Notes, and Comments facilities.
- Time on screen: The program tracked how much time each student spent on each screen.

After completing the program students received a survey which collected reactions to specific software design attributes, the instructional approach, and subjects' perceptions as to how the module effected their learning.

Results

Student Profile

Thirty students participated in the study, 15 of which were male and 15 were female. The majority of participants (83%) were graphic design majors. To obtain a profile of computer experience, students ranked their years of computer use on a scale from 1 to 9 or more years and rated their computer proficiency as "Beginner, Novice, Intermediate or Highly Skilled." The mean number of years of students' computer use was 4 and most (60%) considered their computer proficiency to be at the intermediate level (see Table 1).

Table 1
Profile of Participating Students

Gender	Male 15	Female 15		
Major	Tech. Ed 7%	Graphic Design 83%	Other 10%	
Computer Experience	0-2 Years 30%	3-5 Years 50%	6-7 Years 0%	8-9+ Years 20%
Proficiency	Highly Skilled 10%	Intermediate 60%	Novice 27%	Beginner 3%

Learning Preference

Since the module was the first of its kind introduced to the class, the researchers wanted to know how students perceived it in relation to other instructional approaches. Students were presented 3 statements related to instructional approach: 1) How would they prefer to learn the topic; 2) What is the most effective way for them to learn the topic; and 3) What instructional approaches would you like to see used more. For each of these statements the student had 4 options with which to respond: 1) Lecture; 2) Instruction and assignments; 3) Read a text; 4) A CBI module. It appears that the majority of students prefer to be taught with traditional lecture and assignments and they perceive this approach to be the most effective way to learn a topic (in this case imposition). A large portion of the students felt that the CBI self-instructional approach is something that they would like incorporated into instruction to a greater extent. This finding may result from novelty effects. The students responding just completed the CBI module, and in most cases it was their first exposure to CBI. The novelty of the approach may have affected their reactions. The finding does indicate, to some extent, that students felt favorably about the CBI module.

Table 2
Learning Preference

	Lecture	Instruction & Assignments	Read a text	CBI module
How would you prefer to learn this topic?	7%	48%	0%	45%
What is the most effective way to learn this topic?	4%	50%	0%	46%
What approach would you like to see used more?	0%	25%	0%	75%

Reactions to the CBI Module

The survey presented several statements about the CBI module which students rated on an 8-point scale from 0 (disagree) to 7 (agree). It appears that students liked the CBI approach and saw it as a valuable means to supplement classroom instruction. Of those participating, 100% said they would recommend this program for use next semester and 100% would like to see more modules on other subjects. Moreover, 86% indicated that, to some degree, the module improved their understanding of the topic.

CBI researchers have pointed out that students tend to enjoy this type of instructional material to supplement course work, which may be motivation to assist in their learning (Rattanapian, 1992). The current project appears to support this contention. This is meaningful because the extent to which students desire to learn is a important consideration for those trying to improve educational processes (Ganguly, 1994).

Animation

The module presented animated sequences to model the correct folding of a press sheet. As an exercise, students folded a piece of paper to become oriented to folding direction and page numbering. They could then review an animated sequence to check their fold. Of all aspects pertaining to the program, animation received the highest mean rating, 6.4 out of 7 (see Figure 1). Animated sequences are often effective as attention and motivational devices (Hannafin & Peck, 1988) which gain and maintain student attention. Consequently, students might have reacted positively to them. While the learning benefits of animation are not yet fully understood and this study made no attempt to assess their effects on learning, animated sequences as attention devices may support initial learner processing and thus may be useful (Gagné, 1992).

On-line Glossary

Throughout the module, students had access to an on-line glossary of terms related to the topic. The glossary received high mean ratings (see Figure 1). This is most likely because of the ease at which it could be accessed and the importance terminology and definitions have on understanding imposition (the topic for the module). Of the students using the module, 70% indicated that they used the glossary. Based on the data collected, the importance of an on-line glossary for this content should not be understated.

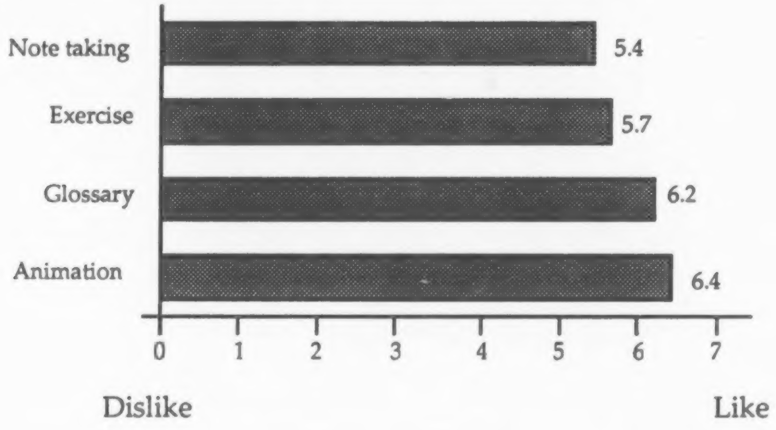
Electronic Note Taking

Electronic note taking enables students to type notes about the lesson into a text area that was saved and later printed for students to keep. Note taking allows students to restructure information in ways which are meaningful, and can serve as an encoding function, thus enhancing learning (DiVesta & Gray, 1972). The note taking feature received a high mean rating (see Figure 1), however, only 30% percent of participants used it. In the researchers' view the context in which the project was tested may have influenced students' use of this feature. Since note taking requires physical and mental effort on part of the student and since no evaluation was given to assess comprehension, students may have lacked incentive to take notes. Note taking as a learning strategy is energetically defended by students (Carrie, 1983) and possibly, under conditions where comprehension is assessed students would use this feature.

Exercise

The module included an exercise in which students labeled the edges of a press sheet. Labeling was achieved by dragging buttons with the computer's mouse to specified locations on the screen. The exercise was incorporated to help students identify press edges and their positions. It required students to become actively involved in the learning process by applying their new knowledge to the task of labeling a press sheet. The exercise received a mean rating of 5.7 of 8, which indicates that student reacted favorably to it (see Figure 1). At this point, exercises of this type appear to have value from an attitudinal perspective and will likely be included in future modules. It is, however, unclear as to whether or not these strategies added any instructional value.

Figure 1
Program Features: Mean Ratings



Student Tracking

The computer-based program incorporated tracking techniques in the software to gain insight into the students' use patterns. The software recorded the number of button clicks, students' antidotal typed comments, and the amount of time spent on each screen. Several observations are made about the process of tracking which may prove useful to researchers, CBI designers and developers.

Depending on the types of data collected, tracking use patterns can yield large amounts of information. From an analysis perspective it is important to keep excess data to a minimum because less time, effort, and resources will be required for analysis. In terms of data storage, it is advisable to store data in an external file. This program was develop in HyperCard and as students went through the module their movements were recorded. To prevent files from becoming too large the data was written to an external file after each student completed the module. It is important that this function should be automated so module use does not have to be directly supervised. External storage also made it easy to separate users' data, preventing a co-mingling of data into one large file.

In this project, students' working paths through the lesson were tracked. The research could see the screens students viewed, in what order, and the number of times. To some extent this information could be used to assess screen design protocols and help determine those designs or features which students used and/or spent time on and those which received superficial attention.

The effects and benefits of tracking techniques need to be addressed in comparative studies which examine it in relation to more conventional data collection methods. As used in this project, the technique appears to be a viable method by which to identifying shortcomings in courseware and to assess students' use patterns.

Summary

The infusion of educational technology into the instructional process for the Graphic Communication class is in its infancy. The challenge is to use technology correctly and to design content in an educationally beneficial manner. The researchers have made a conscious effort to incorporate well designed instruction into media that was thoughtfully chosen. The pilot study seems encouraging in that students have responded positively to this instructional method. What is unknown is how attitudes may change when the module is used as real class content.

Novelty effects are a wonderful thing for curriculum development. The positive test results motivate designers to continue trying, creating, and expanding instruction. But what happens to motivation when novelty wears off? With continued tracking and surveying, the researchers hope that close monitoring will reveal student motivational decline, so appropriate changes in the software design can be made.

References

Carrie, C.A. (1983). Notetaking research: Implications for the classroom. *Journal of Instructional Development*, 6(3), 19-26.

Clement, F.J. (1981). Affective considerations in computer-based education. *Educational Technology*, 21(10), 28-32.

Dence, M. (1980). Toward defining the role of CAI: A review. *Educational Technology*, 20(11), 50-54.

Dick, W., and Carey, L. (1985). *The Systematic Design of Instruction* (2nd Ed.). Dallas, TX: Scott, Foresman, & Co.

DiVesta, F.J., & Gray, G.S. (1972). Listening and notetaking. *Journal of Educational Psychology*, 63, 8-14.

Gagne, R.M., Briggs L.J. & Wager, W.W. (1992). *Principles of Instructional Design*, 4th ed., Orlando, FL: Harcourt Brace Jovanovich.

Ganguly, I. (1994). The motivational effect of televised instruction on teacher directed science learning. In Beauchamp, D., Barden, R. & Baca, J. (Ed.). *Visual Literacy in the Digital Age.* Blacksburg, VA: The International Visual Literacy Association, Inc.

Gleason, G.T. (1981). Microcomputers in education. The state of the art. *Educational Technology*, 21(3),7-18.

Hannafin, M. J., & Peck, K. L. (1988). *The Design, Development, and Evaluation of Instructional Software.* New York: Macmillan Publishing.

Kozma. R.B. (1991). Learning with Media. *Review of Educational Research*, 16(2), 179-211.

Orey, M. A. & Nelson, W. A. (1991). *Using intelligent tutoring design principles to integrate cognitive theory into computer-based instruction.* ED335003.

Rattanapian, V. (1992). *The effects of learner characteristics and an evaluation override option on achievement, attitude and pattern of program use in computerized drill and practice.* Ph.D. Dissertation, The Pennsylvania State University.

Relan, A. (1992). *Motivational strategies in computer-based instruction: Some lessons from theories and models of motivation.* ED348017.

Splittgerber, F.L. (1979). Computer-based instruction: A revolution in the making? *Educational Technology*, 19(1),20-26.

Attitudes of CIS Graduates Toward Course-work Needed For Entry-Level Computing Positions

Gireesh K. Gupta
Department of Computer Studies
Belmont Abbey College
Belmont NC 28012
704-825-6824

Abstract

A survey of Belmont Abbey College CIS graduates was conducted to gather their views on the course-work needed for obtaining & succeeding at entry-level business computing positions. The study revealed the following findings:

i) written communications, speaking skills, and interpersonal skills were rated the most important to perform well at entry-level computing positions, and not the knowledge of computer technology,

ii) the knowledge in the areas of microcomputer application software, logic, and general business was also considered very valuable by the graduates,

iii) Local Area Network technology was strongly suggested to be an integral part of the CIS curriculum.

Introduction

During 1992, I conducted a survey of Belmont Abbey College CIS (Computer Information Systems) graduates and gathered responses. The survey was based on a similar survey of MIS graduates conducted by Dr. Mary Sumner of Southern Illinois University at Edwardsville. Her survey was modified to fit the CIS program's needs and requirements at Belmont Abbey College [see Appendix 2].

This paper presents results of this survey. The survey had two purposes: 1) to determine how well the CIS program at Belmont Abbey College has prepared its graduates and how the CIS program may better serve the community, and 2) to help the Computer Studies Department at Belmont Abbey College in evaluation and revision of the courses and programs it offers.

Belmont Abbey College is a small four-year liberal arts college with a total student population of 1000+ located near Charlotte metropolitan area in the southeastern state of North Carolina. The Department of Computer Studies offers a B.S. degree with CIS major to the adult students in the evening Adult Degree Program and to the traditional Day students. The Department also offers a CS minor and a MIS minor. The Department has approximately 50+ CIS majors.

The survey was sent to sixty-six (66) CIS graduates ranging in graduation dates from May 1986 to May 1991. Twenty-two (22) survey responses were received, thus making a response rate of 33%.

Data Tabulation

The data were tabulated using LOTUS 1-2-3 release 2.0 software package. Table 1 lists the percent responses representing the importance of each CIS area to the graduate's current job responsibilities. In the last column of Table 1, the Very Important or Somewhat Important responses were collapsed to give a clearer picture of the importance of each CIS area. The graphs were created using Quattro Pro Version 2.0.

Results

Analysis of the survey data has revealed the following findings:

1. Most of the CIS graduates who responded were employed in the Charlotte metropolitan area and in the state of North Carolina. Among the respondents, 81.8% were male and 18.2% were female.

   The Charlotte metropolitan area (including small towns and cities within 40 mile radius of the College) employed 68% graduates, other cities in North Carolina employed 18% graduates, and 13.6% graduates were employed in other states.

   81.8% graduates reported that their first job (or first job change, if already employed while attending Belmont Abbey College) after graduation was in the CIS area. Only 18.2% had their first employment in the non CIS area.

   Most (72.7%) of the graduates were employed in the CIS industry working either as computer programmer or systems analyst or programmer/analyst or consultant or systems engineer or in the computer operations area. 9.1% were self-employed in the computing business.

2. 45.5% graduates reported that they obtained second or third employment elsewhere after their first job. 90% of these [45.5%] graduates worked on their first job in the CIS area and 10% in a non CIS area. 70% of these [45.5%] graduates still remained in the CIS area after their first job, whereas 30% worked in non CIS area after their first job.

54.5% graduates did not report any other employment history except their first job.

3. 68% graduates said that the CIS degree from Belmont Abbey College helped them obtain their first job. 9.1% were already employed and continued with their current job after graduation. 22.7% said that the CIS degree from Belmont Abbey College did not help them obtain their first job.

4. 63.6% graduates said that the liberal arts education at Belmont Abbey College helped them obtain their first job. Some of these graduates commented that in addition to CIS skills, the liberal arts education provided them with communication skills, religion skills, and general business skills, which made them a well-rounded person.

   9.1% graduates had already received liberal arts education at other colleges before they transferred to Belmont Abbey College to complete their bachelors degree.

   22.7% graduates reported that the liberal arts education at Belmont Abbey did not help them obtain their first job.

5. The three areas that were rated the most important for performing current job responsibilities were written communications, speaking skills, and interpersonal skills, and not the knowledge of computer technology [see Figure 1]. These areas scored percent responses (when very important or somewhat important responses were added together) between 86.4% to 95.5% [see Table 1]. Similar results were also found in the 1989 study of Sumner et. al. [1], thus reinforcing this Study's finding and demonstrating the continued importance of these areas in successful job performance. Our CIS program prepares graduates in these areas through course-work, team projects, and public speaking.

   Microcomputer Application Software, Logic, and General Business Knowledge were rated the second most important by the graduates for performing current job responsibilities [see Figure 1]. All these areas were ranked at the same importance level of 81.8% (when very important or somewhat important responses were added together) [see Table 1]. Our CIS program requires course-work to educate CIS majors in all these areas, except in the area of Microcomputer Application Software.

   Basic Systems Analysis, Basic Data Communications, Operating Systems Concepts, and Program Testing and Maintenance were rated at the third level of importance (all receiving the same importance level of 72.7% when very important or somewhat important responses were added together) [see Figure 2 & Table 1]. The Computer Studies Department requires two semesters of course work in Systems Analysis & Design area. At the time of this survey, the department offered Networks and Telecommunications course as an elective, and did not offer any course to its majors in Operating Systems and in Program Testing and Maintenance area.

Advanced Structured Systems Analysis Tools, Basic Database Design, and Data Structures were ranked next (all at the same importance level of 68.2% when very important or somewhat important responses were added together) [see Figure 2 & Table 1]. The department requires course-work in all these areas.

Other areas considered important by the graduates for performing current job responsibilities were COBOL programming, Advanced COBOL, Advanced Database Design, Local Area Network Design and Management, Decision Support Systems, Design and Analysis of Algorithms, Basic Accounting, and DP Security and Auditing. These areas received responses between 45% to 64% (when very important or somewhat important responses were added together) [see Table 1]. The Computer Studies department requires two semesters of course-work in COBOL programming, a course in Logic and Algorithms, and a course in Basic Accounting. At present, the department does not offer a course in Advanced Database Design, Local Area Network Design and Management, Decision Support Systems, and DP Security and Auditing.

The CIS areas scoring between 36% to 41% of responses (when very important or somewhat important responses were added together) were BASIC programming, C programming, Programming in 4 GL's, and Use of a CASE Tool [see Table 1].

The CIS areas that scored the lowest responses (between 9% to 28%) were RPG programming, PASCAL programming, and Expert Systems [see Figure 3 & Table 1].

6. Among the computer programming languages, COBOL was ranked the most important language by the graduates for performing current job responsibilities. C and 4GLs were ranked the second most important. BASIC was placed as the third, RPG as the fourth, and PASCAL as the least important programming language [see Figure 4].

7. Additional comments by the graduates emphasized the following areas that were important for successful performance in entry-level business computer positions within their firms: knowledge of operating systems, OS/2 operating system, MS-windows, business writing, interpersonal skills, job control language (JCL), PC applications software, object-oriented programming, CICS (on line programming), COBOL programming, database management, co-op experience before graduation, and strong knowledge of hardware. They also suggested that Belmont Abbey should offer a course in Local Area Network Design and Management, and offer more courses in programming languages such as JCL and Assembly language.

8. In response to the question whether any Belmont Abbey resources played any role in finding their first job after graduation, 4.5% were assisted by the faculty, 4.5% were assisted by the campus placement office, 4.5% were assisted by the campus co-op office, 4.5% were self-employed, 36.4% found their own job without any assistance from the College, and 45.5% did not answer the question.

Conclusions

The Study indicates that the CIS program at Belmont Abbey College has prepared its graduates well and is meeting its goal of preparing application programmers, systems analysts, or programmer/analysts since most graduates are employed in these areas in the CIS industry.

The Study also suggests that our CIS program needs to offer course-work in areas that reflect the current industry. In order to keep up with the changing industry, following new courses have been added to the CIS curriculum at Belmont Abbey as a result of this study: a required course in Local Area Networks, a required course in Program Testing and Verification, and an elective course in Operating Systems [see Appendix 1].

COBOL continues to be a leading programming language (as it is rated significantly more important than BASIC, RPG, PASCAL, or C by the graduates in this study) to find and succeed at entry-level computing positions. This finding is also supported by the Studies of Sumner et. al. [1], Myers [2], and Millhoff [3].

Although not supported by this Study, there are numerous entry-level job opportunities in the Charlotte metropolitan area in AS/400 RPG. This observation is also supported by the Study of Myers [2]. Therefore, an elective course in RPG was reinstated recently by the department.

The knowledge of Microcomputer Application Software remains and will continue to remain very important to find and succeed at entry-level CIS jobs. Although, at present, the Computer Studies Department does not offer a whole-semester course in this area, several industry standard application software packages (such as WordPerfect 5.1, LOTUS 1-2-3 release 3.1) and MS-DOS 5.0 are taught as part of a freshman-level Introduction to Computing course. The Computer Studies Department offers this course as a service course as it is required by most of the academic departments of their majors. Thus, this course satisfies the computer literacy needs of most (if not all) of the College graduates.

Recommendations

For successful job performance in the current CIS industry, the knowledge of Local Area Network was considered very important by many graduates as expressed in their written comments, and should be an integral part of the business-oriented CIS curriculum in two-year or four-year colleges.

This Study indicates that the CIS programs at the nation's colleges need to place heavy emphasis on the written communications, speaking skills, and interpersonal skills. The CIS programs need to educate graduates beyond the traditional technical approach to programming and systems analysis. They need to prepare not only good programmer/analysts, but also good team workers, writers, and potential leaders.

References

1. Sumner, Mary, Robert Klepper, Robert Schultheis. An Assessment of the Attitudes of Graduates and Employers Toward Competencies Needed for Entry-Level MIS Positions. ISECON '90, Chicago, 1990, p. 129-134.

2. Myers, William. The Job Market for Entry-Level Computer Professionals in a Southeastern City. Proceedings of the Sixth Annual Southeastern Small College Computing Conference, 1992, p. 50-58.

3. Millhoff, Brian L. Curriculum Relevance Survey By Pikeville College: Findings And Conclusions For Academia. Proceedings of the Fifth Annual Southeastern Small College Computing Conference, 1991, p. 16-26.

Acknowledgements

I wish to thank Dr. Mary Sumner for allowing me to use her survey as the basis of my survey questionnaire; the Alumni Office and Dr. Artin Arslanian at Belmont Abbey College for their assistance in conducting the survey; Belmont Abbey College for financial assistance in conducting the survey and to do research on this paper; Mrs. Carolyn Pesackis for her assistance in creating the graphs; and the Burroughs Wellcome Fund of North Carolina for a grant to present this paper at this conference.

Appendix 1

Computer Information Systems Major at Belmont Abbey College
(includes changes effective Fall 1994)

I. Core Requirements .... 57 credits

II. Required Business Support Courses .... 18 credits

Principles of Accounting I	(3)
Foundations of Business Administration or Organizations and Management	(3)
Introductory Economics I or II	(3)
Discrete Mathematics	(3)
Statistics for Business & Economics	(3)
Business Communications	(3)

III. Computer Information Systems Requirements ... 42 credits

Intro to Programming	(3)
Computer Organization and Systems	(3)
Programming in COBOL I & COBOL II (3,3) or Local Area Network & C (3,3)	(3,3)
Files and Database Access	(3)
Data Structures	(3)
Information Systems Analysis	(3)
Information Systems Design	(3)
Logic and Algorithms	(3)
Program Testing & Verification (3) or Foundations of Computing (3)	(3)
Management Information Systems	(3)
Applied Software Development Project	(3)
two other CIS electives	(3,3)

Computer Information Systems Electives: *offered on demand

Programming in RPG	(3)
Spreadsheet Applications	(3)
Production and Operations Management	(3)
Advanced Structured Programming	(3)
* Programming with APL	(3)
* Special Topics in Microcomputer Appl	(3)
* Computer Architecture	(3)
* Data Communications	(3)
* AI and Expert Systems	(3)
* Computer-Based Modelling and Simulation	(3)
* Management Science Methods	(3)
* Operating Systems	(3)

IV. General Electives .... 13 credits

Appendix 2

BELMONT ABBEY COLLEGE
Department of Computer Studies

SURVEY OF GRADUATES

Name ______________________ Home Address ______________________

City, State, Zip ______________________ Home Phone ____________

Graduation Date ______________________ Sex ____________

1. First Job (or First Job Change, if already employed) After Graduation

Job in CIS area? Y/N	Mo Year to Mo Year	Job Title	Company	Location	
				City	State
---------	-- ---- to -- ----	---------------	-------------	-----	----

2. Other Employment History (most recent first)

Job in CIS area? Y/N	Mo Year to Mo Year	Job Title	Company	Location	
				City	State
---------	-- ---- to -- ----	---------------	-------------	-----	----
---------	-- ---- to -- ----	---------------	-------------	-----	----
---------	-- ---- to -- ----	---------------	-------------	-----	----

3. Do you feel that the CIS degree from Belmont Abbey College helped you obtain your first job ? YES ( ) NO ( )
Please comment: ______________________________

4. Do you feel that the liberal arts education at Belmont Abbey College helped you obtain your first job ? YES ( ) NO ( )
Please comment: ______________________________

5. Please rate how important each of the following areas is to your current job responsibilities.

Rate: Very important (Very Imp.) Not important (Not Imp.)
Somewhat important (Some Imp.) Not applicable (Not Appl.)

Very Imp.	Some Imp.	Not Imp.	Not Appl.	
☐	☐	☐	☐	COBOL Programming
☐	☐	☐	☐	Advanced COBOL (file Structures)
☐	☐	☐	☐	BASIC programming
☐	☐	☐	☐	RPG Programming
☐	☐	☐	☐	PASCAL Programming
☐	☐	☐	☐	C Programming
☐	☐	☐	☐	Microcomputer Application Software

Very Imp.	Some Imp.	Not Imp.	Not Appl.	
☐	☐	☐	☐	Basic Systems Analysis
☐	☐	☐	☐	Advanced Structured Systems Analysis Tools
☐	☐	☐	☐	Basic Data Communications
☐	☐	☐	☐	Basic Database Design
☐	☐	☐	☐	Advanced Database Design
☐	☐	☐	☐	Local Area Network Design and Management
☐	☐	☐	☐	Decision Support Systems
☐	☐	☐	☐	Expert Systems
☐	☐	☐	☐	Operating Systems Concepts
☐	☐	☐	☐	Programming in 4GL's
☐	☐	☐	☐	Use of a CASE Tool
☐	☐	☐	☐	Logic
☐	☐	☐	☐	Data Structures
☐	☐	☐	☐	Design and Analysis of Algorithms
☐	☐	☐	☐	Program Testing and Maintenance
☐	☐	☐	☐	DP Security and Auditing
☐	☐	☐	☐	General Business Knowledge
☐	☐	☐	☐	Basic Accounting
☐	☐	☐	☐	Written Communications
☐	☐	☐	☐	Speaking Skills
☐	☐	☐	☐	Interpersonal Skills

6. Please comment on any other areas not addressed above that are important for successful performance in entry-level business computer positions within your firm.

____

____

7. Did any of the following Belmont Abbey resources play any role in finding your first job after graduation ? (check all that apply)

Placement office ( ) Faculty ( ) Other ( ) specify: __________

Please comment: ____________________________________

BEST COPY AVAILABLE

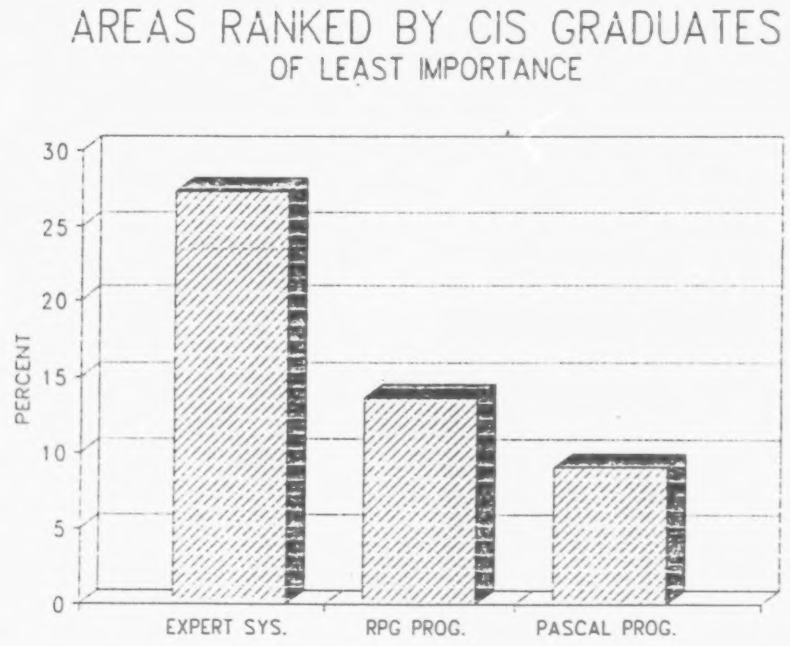

AREAS RANKED BY CIS GRADUATES
OF LEAST IMPORTANCE

FIGURE 3

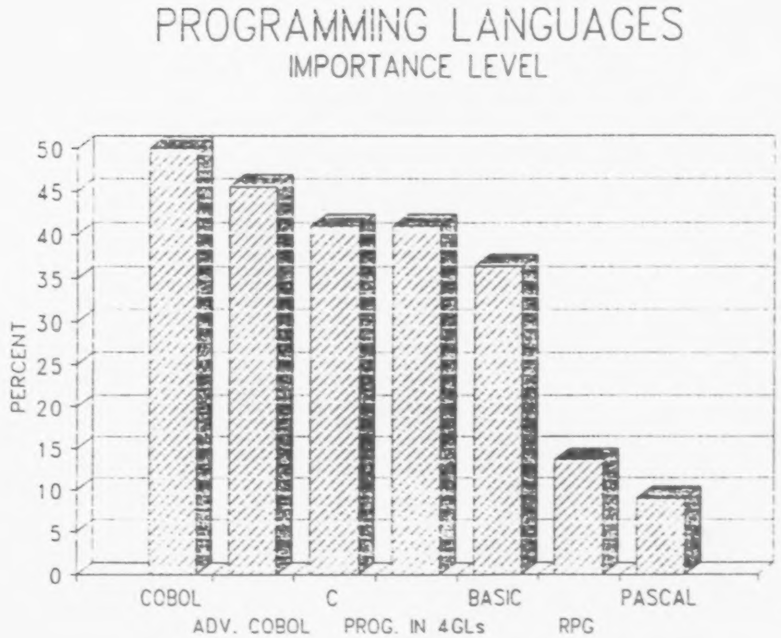

PROGRAMMING LANGUAGES
IMPORTANCE LEVEL

FIGURE 4

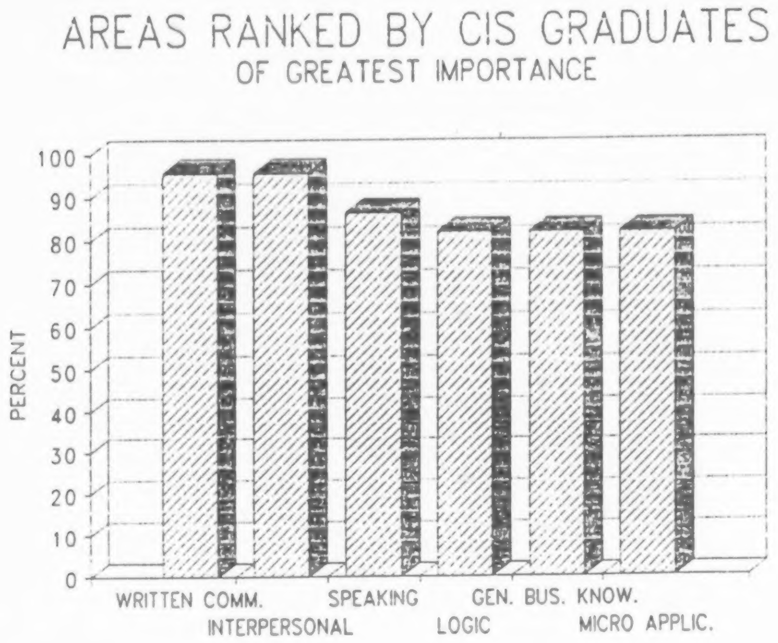


FIGURE 1

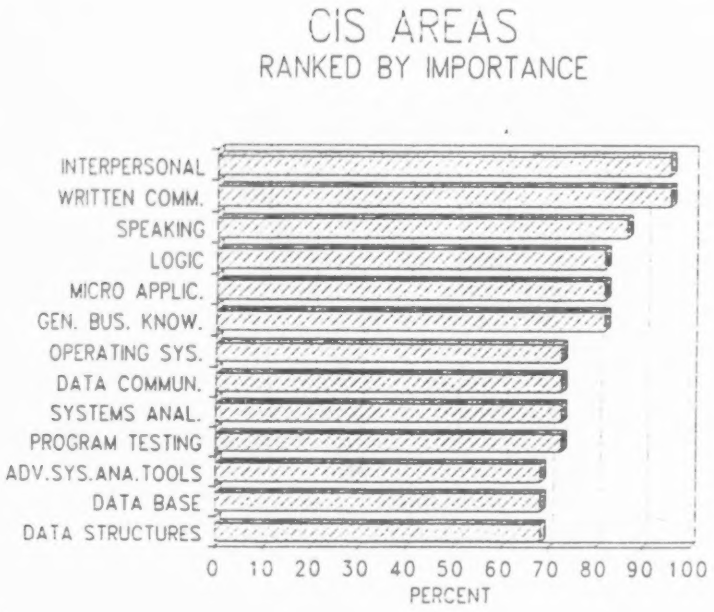


FIGURE 2

CIS AREAS	VERY IMPORTANT	SOMEWHAT IMPORTANT	NOT IMPORTANT	NOT APPLICABLE	VERY OR SOME IMP *
COBOL Programming	31.8%	18.2%	18.2%	31.8%	50.0%
Advanced COBOL (file structures)	31.8%	13.6%	22.7%	31.8%	45.5%
BASIC Programming	9.1%	27.3%	4.5%	59.1%	36.4%
RPG Programming	9.1%	4.5%	18.2%	54.5%	13.6%
PASCAL Programming	4.5%	4.5%	13.6%	63.6%	9.1%
C Programming	18.2%	22.7%	13.6%	36.4%	40.9%
Microcomputer Application Software	59.1%	22.7%	9.1%	9.1%	81.8%
Basic Systems Analysis	50.0%	22.7%	9.1%	9.1%	72.7%
Advanced Structured Systems Analysis Tools	36.4%	31.8%	13.6%	13.6%	68.2%
Basic Data Communications	31.8%	40.9%	13.6%	9.1%	72.7%
Basic Database Design	50.0%	18.2%	13.6%	13.6%	68.2%
Advanced Database Design	36.4%	22.7%	18.2%	13.6%	59.1%
Local Area Network Design and Management	40.9%	18.2%	22.7%	13.6%	59.1%
Decision Support Systems	27.3%	27.3%	18.2%	18.2%	54.5%
Expert Systems	4.5%	22.7%	22.7%	36.4%	27.3%
Operating Systems Concepts	50.0%	22.7%	4.5%	18.2%	72.7%
Programming in 4GL's	18.2%	22.7%	9.1%	40.9%	40.9%
Use of a CASE Tool	9.1%	27.3%	9.1%	45.5%	36.4%
Logic	45.5%	36.4%	4.5%	4.5%	81.8%
Data Structures	36.4%	31.8%	9.1%	13.6%	68.2%
Design and Analysis of Algorithms	18.2%	36.4%	22.7%	13.6%	54.5%
Program Testing and Maintenance	54.5%	18.2%	4.5%	13.6%	72.7%
DP Security and Auditing	4.5%	40.9%	18.2%	31.8%	45.5%
General Business Knowledge	54.5%	27.3%	9.1%	0.0%	81.8%
Basic Accounting	22.7%	40.9%	18.2%	13.6%	63.6%
Written Communications	63.6%	31.8%	0.0%	0.0%	95.5%
Speaking Skills	59.1%	27.3%	4.5%	4.5%	86.4%
Interpersonal Skills	90.9%	4.5%	0.0%	0.0%	95.5%

TABLE 1.

* Very Important or Somewhat Important responses are added together for each CIS Area

Auditing: One Answer to Meeting Faculty Development Computing Needs

Joan M. Hofmann
Department of Child Study/Education/Special Education
St. Joseph College
West Hartford CT 06117
203-232-4571
Hofmannj@ganesha.sjc.edu

Garmston (1993) notes that stories are useful presentation tools for three reasons: 1) they personalize content, 2) they open windows to your intuitive knowledge as an audience, and 3) they can tap the resources of the unconscious mind.

But, what's in a story? There's a setting or context, a plot, one or more themes, characters (usually including a main character), a problem or conflict of some sort (and a resolution to it), and possibly a more general "moral to the story" or some kernels of wisdom drawn from the lived experience. Given these components, I'll share with you a story about my experience--as a faculty member in an education department--of auditing an undergraduate course during the spring semester of 1994 as a means of professional development in the area of computing and technology.

As a case study, this story is based on documents, artifacts, interviews, journal/log entries, field notes, tape recordings, memos, electronic mail transactions and memories. In fact, this story has not really ended. The resolution and the kernels of learning are still being sorted out. But they are not unclear, and will be revealed as the story unfolds.

Like any good story, telling this one raises questions as well as sparks insights. It calls for both the teller and the listener to make sense of the experience. It allows for a good sequel to be written at some point. And so, I invite you to think along with me. In this process we'll 1) acknowledge the role of adult learning, 2) discuss the purposes of professional development and 3) recognize the unique features of auditing in a higher education community, particularly in the area of technology and computing.

Setting

Where did this experience take place? It happened at Saint Joseph College, a small school located in a suburb of Hartford, Connecticut.

The College was founded in 1932 by the Sisters of Mercy as an institution combining professional and liberal arts preparations. It is comprised of the Women's College (a four-year programleading toward an undergraduate degree), the McAuley Weekend College (providing undergraduate programming to a coeducational adult population) and the Graduate Studies Program(providing for graduate and advanced graduate studies). There are approximately 70 full-time faculty and about 275 students graduate from the institution each year. The twelve buildings of the college include a library, athletic center, administration building, chapel, private school for children with disabilities, student union, dormitories and convent. Classrooms are distributed throughout the various buildings.

In terms of technological support, currently the environment includes VAX VMS, IBM token ring, PC's and MacIntosh, and interactive video. We have a small Ethernet network in one building that extends into four rooms. This is connected to the Internet via a gateway to JvNCNet at Princeton, New Jersey. Limited telephone dialing on two lines is available to faculty and students.

There has been an effort in the past three years to provide computers to staff throughout the college. Faculty in some academic departments may have access to a computer located with a staff person in a proximate office. Others have computers in their offices or use laptops. Library resources include automated circulation via the CARL Network, DIALOG, ReQuest, ERIC and CINAHL CD-ROM workstations, and online interlibrary loan via the OCLC Network.

Faculty development has been the focus of increasing attention at the college. A faculty development committee was established in 1992 and a room was recently designated as a Faculty Development Center. The faculty handbook describes various faculty development opportunities, including: coursework availability, consortium activities, monetary support, mini-grants and sabbaticals. Related, each faculty member is required to submit to the Provost an annual faculty development report which details individual goals and efforts in the three areas of: teaching; scholarship and professional activities; and service to the College and community.

Main Character

As the main character in this story, so to speak, let me describe myself as a third-year Assistant Professor in the Department of Child Study/Education/Special Education at Saint Joseph College. I teach in a teacher preparation program at both the undergraduate and graduate levels, focusing in the areas of instruction, curriculum and assessment. I am an active member of the campus community. Among other roles, I serve on the Faculty Development Committee and this year have additionally been appointed to the Provost's Campus Technology Committee and the Education Department's Technology Advisement Committee. My background in the area of computing and technology is novice andnascent; I have more interest and motivation than either expertise or competence. In this respect, I suspect that I am more like, than unlike, the majority of faculty. One of the goals that I noted in my faculty development report this year is "to explore avenues for further developing media and technology in teacher preparation." It is with this goal in mind that I decided to audit the new undergraduate course, "INTD 110: Networking and Communications" during the spring semester of 1994.

Larger Context

But what brought me to the decision to audit a course in telecomputing? In addition to reflecting on my related abilities and competencies, I recognize the role of several inducements, including the influence of information about technology through popular media, voices in the field of higher education, and trends in the field of education and teacher preparation.

It would be difficult to for any of us to dismiss the impact of the constant bombardment of information about computing and technology in the popular press. Not a day goes by that there isn't some related article in the local paper or in the evening news. Al Gore and the Superhighway are alluded to in casual conversation. Information anxiety is now a common experience (Wurman, 1989). Tapscott and Caston describe the phenomena well when they note that: "Global telecommunications networks energize the metabolism of world commerce and move us inexorably toward Marshall McLuhan's global village" (1993, p. 2).

An additional influence in making the decision to audit was my recognition of changes in perspectives on the college curriculum. The enigmas posed by the availability of a superabundance of information and the abilities necessitated by this environment, require leadership in higher education to ensure college graduates' development of a full range of abilities for their future flexibility and success (Breivik & Jones, 1993).

Further, there are significant trends in the field of teacher preparation that were an impetus in my choosing to audit this course. NCATE is the National Council for Accreditation of Teacher Education. In 1989 the International Society for Technology in Education (ISTE) Accreditation Committee was formed to collaboratively develop, with NCATE, standards for teacher education related to computer knowledge and skills. The ISTE maintains that all current candidates seeking initial certification or endorsements in teacher preparation programs should demonstrate a set of competencies relative to the use of computers, telecommunications, and related technologies. (See Appendix for listing of these competencies.) As noted by Abramson:

> [t]hese standards and indicators reflect the acceptance ofeducational technology as a fundamental part of the teaching, learning, assessment, evaluation, and productivity processes. They give credence to the importance of technology across the grades and curricula, to the need for teachers with a wide-range of technology skills and concepts, and to the need for teacher-educators who will remain on the cutting edge of educational technology (1993-94, p. 30).

In light of the factors that I have outlined, it seemed to me that I had work to do both in terms of my own development and in terms of what I would be able to pass along to my students as future teachers. Reflecting on my professional imperatives and personal needs for development, when I heard of the new, undergraduate "Networking and Communications" course, I thought auditing would assist me not only in developing my understandings but possibly help me to upgrade my research skills using telecommunication networks.

The Course

The course I enrolled in was "Networking and Communications: Worldwide Computer Networks" and the syllabus notes that "A major course objective is to comprehend the Internet and its utility in the launching of the National Information Superhighway "(DeGray, 1994). In this two-credit course our focus would be on topics such as: The History and Future of Computer Networks (e.g. Internet), Uses of Networks, Internet Mail, User Services, Remote Logins, Standards Bodies, and Computer Mediated Communication and the Law. In addition to me, there were three students registered for the course. The recommended text, The Whole Internet: User's Guide & Catalog (Krol, 1992),was supplemented by additional texts authored by Kehoe (1992) and LaQuey and Ryer (1993).

The Experience of Auditing: The 'Student' Perspective

As a student, I found the experience of auditing the course to be stimulating and broadening. I learned about both content and process, and I learned more about myself as a learner. For example, I know I learn best in an exploratory atmosphere with an underlying structure. This was reiterated in this course. My journal is filled with notes regarding a) my development of new understandings about telecommunications and Internet throughout the semester, b) my occasional frustration with the pace of the course and organization of lessons, and c) my own difficulties in grasping the content. Throughout the course I wrestled with the stages of the learning sequence that Garmston describes so pointedly:

> -unconscious incompetence(I don't know that I don't know)
> -conscious incompetence (now I know that I don't know)
> -conscious competence (I know as well as my own name and so don't have to put conscious effort into it). (1994, p. 66)

A self-assessment reveals that I am still at the unconscious incompetence and conscious incompetence stages relative to computer/technology content. Through this course, very episodically I caught glimpses of conscious competence as I revisited some of the increasingly familiar territory of electronic mail, subscribing and unsubscribing to lists, accessing remote data bases and the like. Several notebook and journal entries reflect these stages.

This experience repeatedly brought me to wonder about the nature of computing and technology that punctuates the stages of competency development and their accompanying emotion. Perhaps there are some elements related to content phobia similar to that recognized in women in the area of mathematics (Tobias, 1991; Tobias, 1990). Clearly, my journal holds many entries relative to surprise, consolation, and affirmation. Repeatedly, I struggled to understand and manipulate the computer. The words of struggle are clarion. Concomitantly, the feelings of exploration and excitement are evident throughout my notes.

This was clearly an energizing experience, both in terms of the process of learning about and learning to do. The balance and tension of concrete and abstract continued throughout the duration of the course. Can you do before you understand? Can you understand before you do? I suspect that courses in telecomputing cause one to bump into this provocative thorn again and again. I certainly experienced it first-hand.

Auditing was also a valued opportunity to learn with others. The conversations with students before and after class were helpful, enjoyable and stimulating. It was here that one learned from others about newly-encountered lists or ways of making sense of Internet resources. The sharing of one's efforts could be done in a supportive place where it was okay not to fully understand.

Further, auditing allowed me to be very self-directed in my studies. By auditing, I was able to decide how much energy I could devote to this course and also to define the areas of study that would receive my attention. This was a critical dimension to the experience. Undoubtedly, I maintained momentum and motivation by meeting my own needs relative to what I needed to know at the time.

Finally, some logistical matters that I encountered are noteworthy. For example, in the initial class session the course schedule was changed. This re-scheduling was very important to me since meeting once a week was far more 'doable' than two, short sessions. Also, my journal entry reveals the relief I felt as the instructor reviewed the course assignments. I realized the importance of giving and taking from this course according to my own individual needs. Looking back at my journal entries later, I was struck by the importance I attributed to theseissues. They were of paramount concern, on equal footing with the content of the course.

The Experience of Auditing: The 'Teacher' Perspective

While I fully predicted benefiting from auditing the course when I registered for it, my expectations were limited to those I anticipated as a student in the course. Surprisingly, what I encountered in the very first session was the 'teacher in me.' In time I became more comfortable with this familiar inner voice or second level of thinking that would effortlessly take over. There were innumerable times while sitting in class that I would recognize a 'teaching dilemma' or a 'teachable moment' and find myself catapulted into my thinking as a teacher. At times I would be confronted with the projected difference between my own reaction to the precipitating situation and that of the instructor. More often I would take an observer's interest in the strategies that the instructor invoked as he worked with students not unlike my own.

The learning experience extended outside of the weekly meetings. Regularly and unexpectedly, situations from the class would come to mind between class sessions, causing me to reflect on related pedagogical issues. For instance, I frequently wondered about 'content-area pedagogy' relative to technology. Since I am not a teacher of technology, but thinking about the issues of teaching with technology, the class provided a weekly 'field encounter' of sorts.

Additionally, the 'teacher in me' recognized some of the issues inherent in trying to work with even a small group of people who have unique interests and disparate breadths and depths of background knowledge. As a class, we spanned the continuum of expertise with telecommunications, computers and Internet, ranging from novice to more accomplished. Emotionally, many of us were dealing with trying to define the amorphous 'nature and being of Internet,' while others had worked with it long enough to feel okay to 'just go with it.' I fully recognized the challenge of teaching amidst such diversity.

The Next Chapter: Implications for Professional Development and Lessons on Adult Learning

I will now turn to explore some connections to professional development at the college level and adult learning. Lastly, some recommendations will be put forward.

The urgency for providing opportunity for the development of faculty competency in all of the traditional areas of faculty professional development (teaching, research/publishing/presenting, and community service activities) is undeniable when we acknowledge the importance of faculty toour college institutions. Jennings, Barlar and Bartling note:

An institution's most valuable resource is its faculty. The faculty determine the structure and quality of the curriculum, control the quality of instruction, position the institution relative to creativity and scholarship, and implement the institution's service linkages and relationships (1991, p. 147).

Many are describing the technological paradigm shift and anticipate its far-reaching impact on how we as instructors will teach and how our students will learn. Hypermedia computers connected via international broad-band networks combined with hypermedia authoring and delivery systems will profoundly affect us in the days ahead as we create customized teaching materials, publish electronically and conduct research via remote sites. But getting from 'here' to 'there' will not be easy; in terms of academic budgets, this so-called "technology paradigm shift" couldn't be happening at a worse time (Jensen, 1993).

The call for comprehensive professional development planning is further amplified when we recognize several additional factors impacting our colleges at this point in time: older and less mobile faculty, student demographics, financial constraints, and societal changes (Mott, 1994). These changes include the need for college graduates with computing awareness, readiness and literacy. As we instruct at the postsecondary level we must incorporate new instructional and research methods into our repertoires in order to prepare these students.

While our perceptions and satisfaction with faculty development programs may be quite mixed and somewhat related to the type of institution in which we work (Jennings, Barlar, & Bartling, 1991), there are some shifts in thinking for those considering the importance of enhancing instructional skills. Although the Carnegie Foundation for the Advancement of Teaching report indicates "that the scholarship of teaching is valued less than the scholarship of discovery in U.S. higher education" (1991, p. 30), others report findings such as Fideler: "[t]he recent and belated appreciation for the scholarship of teaching among higher education opinion leaders...is particularly encouraging...."(1991, p. 198). On many campuses, there appear to be visible signs of an increased focus on instruction.

Beyond instruction, faculty interest in telecommunications is induced by far-reaching opportunities for journal and document research from remote sites, collaborative research with colleagues at both neighboring and distant institutions, and communication via bulletin boards and discussion groups with professional organizations and individuals world-wide. Additionally, many organizations are now announcing upcoming conferences via electronic bulletin boards and receiving articles for publishing via electronic mail.

As we attempt to expand our instructional and researchrepertoires, it is important to recognize that integrating innovations requires extended time for adjustment (Guskey, 1990) and change in practice "takes considerable time and is the result of staff development that is conducted over several years" (Wood, 1993). Taking a long-term view is further acknowledged in the Carnegie Foundation Report for the Advancement of Teaching (Boyer, 1990) which recommends developing three to five year professional development plans toward meeting individual and department goals.

The need for greater facility, understanding and integration of computing methodology extends to all fields. For those faculty whose primary field is not computing and telecommunications, access to equipment and expertise on an on-going basis, in a supportive environment, with instruction crafted to match individual needs are a necessity. Auditing of telecommunications courses on the local campus may be one answer.

As adult learners characterized by Apps (1992): we bring to educational settings a set of known and unknown expectations, needs, deficiencies, and assets; we may experience psychological, social, physical and economic barriers to learning; we are multifaceted and present as 'bundles of contradictions' (self-assured and confident, but highly dependent; clear but confused; wanting the teacher to provide the answers, but wanting to search on our own); we appreciate flexibility in teaching strategies; we contend with multiple roles as learners, employees, parents, spouses and the like; we generally want a say in what we learn; and we may know little about a subject but often know more than we realize. Further, we often want to be participants in a process that "is characterized by elements of collaboration, challenge, reflectivity, action, respect, freedom and equality" (Galbraith, 1991, p. x).

While these characteristics of adult learners prompt a variety of perspectives about how to teach adults (Hayes, 1993), they also offer a synthesis of understanding of adult learners that can serve as a basis for planning and implementing staff development (Caffarella, 1993). I will discuss the auditing experience in light of each of the descriptors offered by Caffarella (p. 55):

Adults will commit to learning when the goals and objectives are considered realistic and important to them. The choice to audit clearly allows for an individual to set reasonable learning objectives that are matched to one's motivation and interest. Further, over the course of a semester, these goals can be adjusted to accommodate exigencies that arise or unexpected discoveries that spawn new directions to be explored.

Adults will learn, retain and use what they perceive is relevant to their personal and professional needs. The auditing experience allows for unequivocal self-directedness since essentially all elements of the learning process areself-designed. Be ready, though, to be confronted by your own internal and self-imposed constraints and limitations.

Adult learning is ego involved. Undoubtedly, the auditing experience will provide some tricky navigating as you transect teacher and student roles. Your risks can be as public as you make them. In any event, the interplay of technology and our related reaction to it is far from a direct course: "It weaves and bobs and lurches and sputters" (Naisbitt, 1982, p. 41).

Adults need to see the results of their efforts and have feedback on how well they are doing. Regular class meetings provide a built-in system for self-assessment and monitoring. If you appreciate having deadlines as a means of keeping you on track, the sessions will provide a weekly or bi-weekly 'alarm clock' to assist you. On the other hand, if you prefer the process of learning without having to provide products of that learning, you can enjoy the latitude that auditing affords. In short, you make it what you want, as demanding or as relaxing as you desire. Feedback is available to you in a flexible manner from the instructor, depending on whether you solicit it or make it known that you are a 'quiet' participant. Importantly, feedback can come from other students as well as the instructor.

Adults who participate in small groups are more likely to move their learning beyond understanding to application, analysis, synthesis and evaluation. In addition to the group process during classtime, the auditing experience offers the opportunity for before and after class interaction with other students. Dialogue can spill over beyond the classroom, hallways and computer lab, particularly through the use of electronic mail which makes it possible to easily communicate on a regular basis with the instructor and fellow students.

Adults come to learning with a wide range of previous experiences, knowledge, self-direction, interests and competencies. An auditing experience in a telecommunications course will highlight this diversity among learners and allow it to be used to advantage. In a course in which the Internet with all its rich territory is available for exploration, interests can be pursued and shared while competencies are developed.

Adults want to be the origin of their own learning and will resist learning situations that they believe are an attack on their competencies. No problem here. Auditing permits the flexibility to re-direct and discover at a variety of levels of competency. In the area of telecomputing, the development of competency is never-ending and therefore invites respect along a continuum of learning.

Because the transfer of learning is not automatic for adults, it must be planned for and facilitated. Depending on the motivation for auditing a particular course, the transfer of learning can be nurtured and promoted from the beginning of the semester. If desired, auditing permits you to singularly direct efforts toward the transfer of new understandings to other situations. You are working to satisfy yourself. Further, due to the on-going natureof your relationship with your faculty colleague, there is a very real possibility for support at a later date when you actually attempt to practice or integrate your learnings into new situations.

Based on my experience, I would recommend serious consideration be given to auditing as a means of professional development. If you decide to give this option a try, here are some suggestions:

Recognize the influence of your enrollment on the instructor. It is important that you select your course with thoughtful recognition of the feelings of the instructor and courteously speak with the person in advance. I selected a colleague with whom I had worked before. Our roles as 'expert' and 'novice' were previously established so that when I approached him about auditing the course, he playfully responded, "Oh, I thought you might co-teach it with me!"

Be aware of the impact of your presence on both the instructor and the other students. I still recall a faculty co-auditor apologizing to the class, "Oh, I'm supposed to be quiet. You're the student. I'm the audience." While I do not think this one instance had any real effect on the class, multiple cross-commenting could distract from the learning.

Carefully consider your goals and 'agenda' and, if appropriate, attempt to clarify these with the instructor ahead of time. I very much wanted to cause minimal concern to my faculty colleague as he taught the course for the first time. But, given his accommodating nature, I suspect that I may have been able to meet more of my personal objectives in taking this course if I had explicated them and worked on them with him more collaboratively.

Have reasonable expectations of your capacity to be in class for the scheduled sessions and work this out with the instructor. Given other commitments, I knew in advance that I had a few conflicts with the class schedule at the start of the semester. Yet, even with the best of intentions it was quite difficult to continue to follow through on my commitment to this course. Auditing releases you from the demands of submitting assignments, but a desire to be a part of the 'community of learners' and the need to fulfill responsibilities to the instructor and the other students weigh toward regular attendance and participation.

Be sensitive to the risks that both you and the course instructor are taking. As an auditing student, you are revealing your lack of knowledge and understanding while s/he is revealing the depth of his/her knowledge and teaching expertise. Teachers have been described as social, artistic and enterprising(Holland, 1973). Their work, however, is often individualistic and conducted in private (Goodlad, 1982; Lortie, 1975). As an auditor you are a student rather than a peer coach, observer or mentor. If you wish to explore these other roles, you would want to fullydiscuss the reciprocal roles and mutual benefits beforehand (DeZure, 1993). As an auditor, stick to your role unless you have mutually decided otherwise with the course instructor.

To gain maximum benefit from auditing, maintain a journal and a system of note taking to chronicle the content and process dimensions of your experience. A journal provides a tool for clarifying ideas and experiences. You can return to it and revisit your experiences and refine your thinking. As noted by Holly (1989), " [t]hrough the journal-keeping process, we can become more sensitive observers, more penetrating in our inquiry into 'what it all means,' and more focused on our roles and directions in life (p. 266)."

Enjoy the process of learning. It can be exhilarating to explore new areas, and a real pleasure to experience this with students. As a college professor, if you enjoy teaching young adults you may well relish the process of learning with them as a fellow student.

Professional development is critical to our on-going satisfaction and performance as faculty in post-secondary institutions. As inquiring professionals, we seek out opportunity to enhance our understandings, nurture and challenge our attitudes and beliefs, and develop our skills and competencies. In order to do this, we need a milieu of professional growth and improvement that fosters our self-directed learning and provides us with the resources for on-going development.

To enable ourselves to meet continuing demands and changes, we need to take advantage of a variety of resources and opportunities. Some of these may well be right on our own campuses. Without allocating additional financial resources, an institution can provide faculty access to up-to-date, stimulating opportunities to enhance their professional development through the auditing of telecommunications courses. Such experiences can be gained on local equipment and with proximate colleagues, promoting on-going networking and campus-wide collegial relationships. Auditing may be one way to assist faculty in creatively expanding their instructional repertoires and research capabilities through rewarding experiences that meet their evolving professional needs.

Special thanks to Dr. Ronald DeGray for his hospitality and support.

Appendix

The International Society for Technology in Education (ISTE) maintains that all current candidates seeking initial certification or endorsements in teacher preparation programsshould be prepared to:

1. demonstrate ability to operate a computer system in order to successfully utilize software.
2. evaluate and use computers and related technologies to support the instructional process.
3. apply current instructional principles, research, and appropriate assessment practices to the use of computers and related technologies.
4. explore, evaluate, and use computer/technology-based materials, including applications, educational software and associated documentation.
5. demonstrate knowledge of uses of computers for problem solving, data collection, information management, communications, presentations,and decision making.
6. design and develop student learning activities that integrate computing and technology for a variety of student grouping strategies and for diverse student populations.
7. evaluate, select and integrate computer/technology-based instruction in the curriculum of one's subject areas) and/or grade levels.
8. demonstrate knowledge of uses of multimedia, hypermedia, and telecommunications to support instruction.
9. demonstrate skill in using productivity tools for professional and personal use, including word processing, database, spreadsheet, and print/graphic utilities.
10. demonstrate knowledge of equity, ethical, legal and human issues of computing and technology use as they relate to society and model appropriate behaviors.
11. identify resources for staying current in applications of computing and related technologies in education.
12. use computer-based technologies to access information to enhance personal and professional productivity.
13. apply computer and related technologies to facilitate emerging roles of the learner and the educator.
    (ISTE, 1993, p. 7)

References

Abramson, G. W. (Fall-Winter, 1993). Technology and the teacher education accreditation process. ED-TECH Review: International Forum on Educational Technology Issues and Applications, 27-30.

Apps, J. W. (1992). Adult Education: The Way to Lifelong Learning. Bloomington, IN: Phi Delta Kappa Educational Foundation. Fastback Number 334.

Boyer, E. L. (1990). Scholarship Reconsidered--Priorities of the Professoriate. Princeton, NJ: Carnegie Foundation for the Advancement of Teaching.

Breivik, P. S., & Jones, D. L. (1993). Information literacy: Liberal education for the information age. Liberal Education, 79,1, 24-29.

Caffarella, R. S. (1993). Facilitating self-directed learning as a staff development option. Journal of Staff Development, 14, 2, 30-34.

Carnegie Foundation for the Advancement of Teaching. (March-April, 1991). The payoff for publication leaders. Change, 23, 2, 27-30.

DeGray, R. (1994). Course syllabus: INTD 110 Networks and Communication, Saint Joseph College, West Hartford, CT.

DeZure, D. (September-October, 1993). Opening the classroom door. Academe, 27-28.

Fideler, E. F. (1991). Inquiry-based faculty development. Journal of Staff, Program, and Organizational Development, 9, 4, 197-203.

Galbraith, M. (Ed.). (1991). Facilitating Adult Learning: A Transactional Process. Malabar, FL: Krieger.

Garmston, R. (1994). Presenting to groups experiencing change. Journal of Staff Development, 15, 1, 66-68.

Garmston, R. (1993). Tell me a story. Journal of Staff Development, 14, 3, 60-61.

Goodlad, J. I. (1982). Let's get on with the reconstruction. Phi Delta Kappan, 9, 19-20.

Guskey, T. R. (February, 1990). Integrating innovations. Educational Leadership, 11-15.

Hayes, E. (1993). Current perspectives on teaching adults. Adult Education Quarterly, 43, 3, 173-186.

Holland, J. I. (1973). Making vocational choices: A theory of careers. Englewood Cliffs, NJ:Prentice-Hall.

Holly, M. L., & McLoughlin, C. S. (1989). Professional development and journal writing. In M. L. Holly and C. S. McLoughlin (Eds.), Perspectives on teacher professional development. Philadelphia, PA: Falmer Press.

ISTE (1991). Curriculum guidelines for accreditation of educational computing and technology programs. OR: International Society for Technology in Education.

Jensen, R. E. (July-August, 1993). The technology of the future is already here. Academe, 8-13.

Jennings, C. M., Barlar, A. D., & Bartling, C. A. (1991). Trends in colleges' and universities'

faculty development programs. Journal of Staff, Program and Organizational Development, 9, 3, 147-154.

Kehoe, B. P. (February, 1992). Zen and the art of internet.

Krol, E. (1992). The whole internet: User's guide & catalog. Sebastopol, CA: O'Reilly & Associates, Inc.

LaQuey, T., & Ryer, J. C (1993). Internet companion: A beginner's guide to global networking. New York: Addison-Wesley Publishing Co.Lortie, D. E. (1975). Schoolteacher: A sociological study. Chicago: The University of Chicago Press.

Mott, M. C. (1994). REFRAME: A model for promoting faculty/staff vitality and excellence. Journal of Staff, Program and Organizational Development, 11, 3, 151-157.

Naisbitt, J. (1982). Megatrends. NY: Warner Communications.

Tapscott, D., & Caston, A. (1993). Paradigm shift: The new promise of information technology. New York: McGraw-Hill.

Tobias, S. (1991). Math mental health: Going beyond math anxiety. College Teaching, 39, 3,91-96.

Tobias, S. (1990). Math anxiety: An update. National Academic Advising Association Journal, 10, 1, 47-50.

Wood, F. H., & Thompson, S. R. (1993). Assumptions about staff development based on research and best practice. Journal of Staff Development, 14, 4, 52-57.

Wurman, R. (1989). Information Anxiety. NY: Doubleday.

Mobile Computing at Grove City College

Frederick J. Jenny
Grove City College
100 Campus Drive
Grove City PA 16127-2104
412-458-2071
jenny@gcc.edu

Introduction

The ability to use a computer is now expected in all professions. The experience of regular, extensive, and convenient use of a computer will assist students in whatever career they choose. Therefore some institutions have required PCs for incoming students; others have "wired" the campus. How do institutions best prepare students for careers and life in the 21st Century?

Grove City College has chosen to implement a mobile computing strategy. The Class of 1998 will be "issued" personal notebook computers upon arrival for freshmen orientation. Grove City will be one of the first institutions of higher education to equip every student with the ability to communicate from any point on campus, or at home for that matter, to networks on campus and worldwide -- a campus without geographical limitations (or a campus without bounds).

Grove City College is Christian liberal arts college of approximately 2200 students situated in Western Pennsylvania half-way between Pittsburgh and Erie. In the past academic computing at Grove City has evolved in respects not unlike many schools of its size. The following time line of that evolution is given in order to give some perspective to the current computing initiative:

Computing History

TIME	SYSTEM(S)	USER POPULATION
< 1981	DEC PDP 11/70	Administration; a few technical majors
1981	DEC VAX 11/180 added	More academic users
1984	Weir C. Ketler Technological Learning Center opened with 100 DEC Pro 350 IVIS stations on a net.	Large public access facility
1985-6	VAX 11/780 upgraded to 11/780-5; VAX 8650 replaces PDP 11/70	VAX cluster for student & administrative use
1987	Campus voice and data cabling plant upgraded; AT&T System 85 phone switch installed; AT&T SIN installed for terminal and print communications	
1989	PC and Mac desktop computers for faculty	Convenient faculty computing
1990	PC and Mac systems in public access facility: 45 PCs, 50 disciplines Macs, 45 VAX terminals	Usage grows across many
1993	VAXen 8650 & 11/180-5 replaced by VAX 6520	
1993	Planning begins for mobile computing - additional networking	
1994	March: faculty receive notebook PCs and printers September: entering freshmen receive notebook PCs	

Overall Plan

Initially the Grove City College Mobile Computing Initiative is designed to equip users with notebook computers and to work toward the refinement and expansion of the existing network services. The motivation for consideration of a major computer initiative as described is two fold. The existing public access facility (the Ketler Technological Learning Center, or TLC) is barely able to meet the ever growing need for computing power. The local area network of fifty Mac Plus microcomputers, once two-floppy systems with 1 MB RAM, now has 4 MB/ 40 MB hard drive nodes. The 45 Zenith 386/20s on the Novell LAN have been upgraded to 4 MB RAM, DOS 6.2, and Windows 3.1. It continues to be difficult to keep these labs serviceable in terms in the number of workstations, and in terms of the growing hardware demands of system and application software. Putting state-of-the art computing in the hands of incoming students minimizes (not obviates) the necessity of maintaining a public access facility.

A more compelling reason for this type of initiative is to equip each student with the knowledge, tools, and skills necessary for continued life-long learning in this information age. Contemplate the difference in experience and level of technical sophistication, after four years, of two undergraduate students, one of whom uses the computer in public access labs for the necessary assignments and papers, etc.; the other literally lives with a system, used for personal and educational productivity, during the semester, over breaks, during the summer, even during internships. There is a marked difference in performance now between those students who own their own computers and those who do not. (Students and parents realize the necessity of having convenient access to computers. A survey of this year's freshman class at Grove City reveals that approximately one- third of the class came to campus with their own microcomputer.)

Why a notebook computer?

Many institutions have chosen to provide each student with desktop systems -- and with great success. The notebook computer, however, offer several advantages over the desktop system. Its portability makes the computer constantly available, not tied to a central computing facility or dorm room. The notebook computer affords the luxury of having a computer lab everywhere -- in the classroom, in the science lab, in the library. Again, observation of student study patterns at Grove City reveals that more and more students study together in a variety of locations, away from their rooms and with access to some computer.

The Network

The notebook computer has great educational and productivity value in having each student with their own computer even without a network. A 16:1 student:computer ratio becomes 1:1. Planning and evaluation of network services continues. The College has several isolated LANs in buildings across campus, including the TLC, Buhl Library, the science and engineering buildings. Fiber cable that can potentially be used as a campus backbone is in place. Primary goals for networking include increasing communication capabilities, providing access to library resources, and providing access to Internet. Emphasis on the development of the campus-wide network will be given over the next year. In the meantime it is most likely that access to campus resources will be delivered by means of notebook internal modems.

Which Notebook Computer?

Choosing a particular notebook product is a difficult task, a task that may need be faced each year as the technology changes. After much deliberation the College decided on the feature set of the notebook system. The College desires to use a first tier manufacturer, one that has a national reputation for quality, reliability, and financial security. A first tier manufacturer possesses a certain name recognition, in spite of the gain in popularity of generic or second and third tier machines. In fact, the attractive price point of second and third tier machines is offset by additional maintenance costs required to provide the desired service coverage (four years). The first tier manufacturers offer a three-year product warranty. (The College is adding a fourth year through an external organization, rather than perform self-maintenance.)

Other features include Intel 486 processor, large trackball pointing device, an excellent keyboard, NiMH battery with good longevity, color display, 4 MB RAM, 120 MB hard drive, MS-DOS and Windows. Subnotebooks were ruled out because of the external floppy drives. Monochrome systems were originally considered because of the cost of color systems. Reduction in the price of dual-scan passive matrix systems made color viable.

The System

The following describes the system chosen for the 1994-1995 school year:

MODEL	Compaq Contura 4/25c
Processor	Intel 486SL-25
Memory	4 MB RAM
Storage	120 MB hard drive 1.44 MB 3.5-inch diskette
Display	Dual-scan passive matrix color LCD
Pointing Device	Large Compaq (Logitech) Trackball
System Software	MS-DOS 6.2, Windows 3.1
Application Software	MS Works for Windows 3.0
Modem	internal 96/24 FAX modem

The Cost

The cost of this program is being met by extended payment charges of $325 fee each semester or by outright purchase of the system. Incoming freshmen have the choice of having the $325 fee added to each semester's bill, or by purchasing the system outright for approximately $2300. Continuing students have the option of participating in the new program by either purchasing a system outright, or by making an up front payment ($325 times the number of semesters attended as of May 14, 1994) and then paying subsequent $325 each semester. In any case, after the equivalent of eight payment charges have been made, the student owns the computer.

The initial computer acquisition of 600 notebook computers for incoming freshmen constitutes a significant financial outlay. An external organization is handling the financing of this expenditure.

At this point in time, no decision has been made concerning printing services other than to allow students to acquire their own printers. The price point for the desired level of quality of printing is too high to provide in addition to the notebook system. Public access areas will be equipped with port replicators or docking stations to allow students to use existing laser printers on the LAN.

Training

Members of the faculty have been constant users of desktop systems for a number of years. Some, however, have been long-time users of Macintosh systems. With the delivery of Compaq Contura 425c notebooks systems to all faculty, training for the faculty with their new systems has commenced and will continue this summer, where needed. Faculty training is available in the following areas for those who wish to participate:

Computer Basics
Word Processing (Levels I - III)
DOS (Levels I & II)
Application Programming
FAX-Modem Usage
Windows (Levels I & II)
Spreadsheet (Levels I - III)
Presentation Software (Levels I & II)
User Applications
Mac to DOS file conversion

C-CUE, the regional computing consortium to which Grove City belongs, recently signed the Education Select agreement with Microsoft. Thus Microsoft Office will be provided for those faculty who desire it. Further plans for integration of computing into key areas are being finalized.

In the fall of 1994, each entering freshman will receive a personal notebook computer. Through this program, each student will be provided direct accessibility to state-of-the-art computing technology, and the latest methodologies in report and presentation generation, word processing, and mathematical/data analysis. One of the more dominant principles behind the issuance of the computers to incoming students is to lay the foundation for success, not only for the student's academic career, but for his or her professional career as well. It is essential that a sufficient amount of training be provided to each computer user to allow a smooth transition to the point where the computer is an integral part of the overall academic experience. An outline for a six-part plan for providing training to begin to meet this overriding need is as follows:

- Basic Operation of the computer
- Windows Basics
- Word Processing Tools
- Remedial Work
- Utilizing the Computer Operating System
- Spreadsheet/Mathematics Tools

The above plan is expected to be completed in five presentation hours, with built-in session for those students who may need some extra work and hands-on experience. Training will take place in informal classroom type settings with numerous student assistants. Each session will be accompanied by a manual. A full operation (24 hour) 'help desk' will be begin during this time. Ample opportunity will be provided between sessions in order to allow the student time to experiment and more fully utilize his or her computer.

One of the more pressing issues is the allocation of personnel to this training program. If class size is limited to 30 students, it will take 20 classes to hand the freshmen class. To handle this teaching load, it would take approximately 10 faculty (each teaching two sessions) and then approximately 20-50 student assistants who will roam about offering personal assistance in the event of problems.

Support Staff

This special training is in addition to the normal types of support services provided by Academic Computing. Presently Academic Computing staff consists of three full-time professionals in addition to the student workers. Additional staff will be added to manage the 'help desk.' Additional staffing is being considered as plans to expand the campus network evolve.

Conclusion

Grove City College is looking forward to an exciting Fall with the arrival of 600 eager freshmen and their notebook computers. Many challenges and decisions yet face our campus community as we now evolve into a fully mobile computing environment.

Computer Applications In The Physics Curriculum

H. R. Kobraei and R. C. Etherton
Murray State University
Murray KY 42071
502-762-3011

In 1989, two new types of course study were introduced in the curriculum of the physics department. These were Advance Problems in Physics and Engineering Physics, I and II. It was thought that these courses would fill in the gaps of previous learning experience and would integrate this knowledge into a comprehensive work base for students. This work base then allows the students to go beyond the typical text book problems and encounter problems which can not be solved through analytical solutions but require a higher level of solution techniques and applications. These types of problems require the use of tools, such as higher math, approximation techniques, etc., which are usually not available to undergraduate students. However, with today's advances in computer technology and power, students have a source that is readily available to them. The knowledge and technology that is at the students' disposal allows them to design their project, implicate it, and complete it. This is a valuable skill within itself.

In order to assure that these objectives are being met, a series of assignments had to be developed for each course. The courses were broken down into two segments, classical physics and modern physics. In Advanced Problems in Physics and Engineering Physics I, which is usually offered in the fall semester, the assignments included topics in Mechanics of Point and Rigid Systems, Electricity and Magnetism, and Optics. In the second semester class, Monte Carlo Simulation, Thermal Physics, ideal and non-ideal gas systems, Quantum Mechanics, and scattering of particles are the topics of analysis.

The courses, Advanced Problems in Physics and Engineering Physics I and II, are considered to be among the elective courses for our physics and engineering physics majors. A typical student, who participates in these classes, has a junior or senior status with high class standing. Each course includes ten assignments which have to be completed by each individual student enrolled in the class. Seven to ten days is a typical time for completion of each assignment. After completion of each project, students are required to turn in an assignment report with three distinct parts. Each report must include the introduction to the assignment, the statement of the problem, and special cases where analytical treatment may exist. The introduction also contains the rationale of doing the problem. In the second part of the report, students describe how they have implemented the solution. This may include the numerical techniques, and application of any physical and mathematical approximations. Any test which shows the integrity of the techniques and the actual computer code used is also included in this part. In the last part of each report, the solution and its implication are discussed. In addition to the conclusion, some tips and general problems encountered in the assignment are mentioned.

Each of our classes meet three times per week. In the first meeting , we usually discuss the assignment of the week and supply an introduction to the subject for students, which takes the form of regular lecture. In the second meeting of the week, the possible analytical solutions as well as numerical techniques are discussed. Finally, in the last meeting of the week, we discuss individual problems encountered in solving the problem, and review any helpful hints which may be useful in completion of the project. Along with this, many office hours are spent in helping and discussing the assignments on a one to one basis. The material for the assignments were gathered from several sources. We get the subject for each of the assignments from the text books. We prefer to use books the students have already had in previous courses. This serves as a helpful reminder, since students are already familiar with the notation and with the contents of the book. These books include the current text books that are used in mechanics, electricity and magnetism, thermodynamics, modern physics, quantum mechanics, and statistical mechanics. The numerical and computational techniques are put together from various books in numerical analysis and computational physics. We have used Computational Physics, by Koonin and Meredith, 1990, Introduction to Computational Physics, by De Jong, 1991, Computer Simulation Methods in Theoretical Physics , by Heermann 1986, and Numerical Recipes, by Press, Flannery, Teukolsky, and Vetterling, 1985. The two books by Gould and Tobochnik, Computer Simulation Methods Part 1 and 2 , 1988, have also been greatly utilized in our courses.

In our department, we have a computer laboratory which contains 17 AT, 386, and 486 IBM compatible computers. In addition, we have a HP 730 work station, two Mackintosh II-CI, and a Micro-Vax II with four terminals. As a result, students can work with different operating systems, thereby giving them an opportunity to become familiar with different systems that may be encountered in the future. Students have the liberty of choosing among BASIC, FORTRAN, C, and Pascal programming languages. However, we strongly encourage our students to use Mathematica as a programming language. We have made Mathematica available in all operating platforms (DOS, Windows, UNIX, Mac OS). Mathematica can be used as a high level programming language. The codes written in Mathematica are very short and easy to follow. Furthermore, the time students spend in developing computer codes is drastically reduced. This in turn gives the students the opportunity of spending more time on the project and its solution than on writing a very lengthy computer code.

We have organized the set of problems for Advanced Problems in Physics and Engineering Physics, I and II, in such a way that shows the unity of physics as a field. The assignments cover almost all major undergraduate courses that students have had. As a result, the assignments correlate these courses that students have already taken and unify them. Because of computer hands-on experience, these assignments deepen the students' understanding of subjects, even on concepts that should have been already learned.

To illustrate the experiences of the students and ourselves, it is helpful to mention an assignment from each class. One of the early assignments in Advance Problems in Physics and Engineering Physics I, is a projectile problem. It is about throwing two objects with the same initial angle and velocity but which have different masses. Students are asked to do three parts, (1) the ideal case in which only gravity has an effect on the objects, (2) gravity with the added influence of air friction, and (3) the combined effect of gravity, air friction, and wind force. The students expected that in part 2 and part 3 that there would be a difference in landing location for the objects. However, we observed that some students were surprised to see that in part 1, both objects landed at the same location, although this is one of the most fundamental concepts of physics. Starting with formulating the problem, writing the computer code, and finally, watching the motion of objects on the computer screen has fulfilled many of our expectations that the students would gain a comprehensive understanding of what they are studying.

In the second semester, the assignment we have chosen to illustrate has to do with the kinetic theory of gases. The students are asked to start with one-hundred particles randomly situated in a cubic box with initially random velocity. We have asked the student to do three parts in this assignment. These are, (1) integrate the problem for non-interacting particles (2) solve time evolution of the system for particles which interact via of Lennard-Jones potentials, and (3) integrate the system for ionic particles. We ask students to find the average and mean velocities, energies, temperatures, pressures, etc., of the system. The fact that the students themselves can see how they can relate the microscopic world to the macroscopic qualities, such as pressure and temperature, is very educational for them because it makes it more tangible. As well as the students will see and operate different methods of computer simulation. They do both Molecular Dynamics and Monte Carlo Simulation Method in the second semester.

These two classes are very demanding. They require more time from students and teachers than normally expected. Therefore, it could be a hinderance for those with a heavy load. By utilizing previous materials and sources, students are forced to relearn and retain previous information. This in turn integrates the reinforced knowledge with the new techniques of application creating an appreciation and understanding of physics as a unified field of study. Along with this appreciation and understanding, valuable skills are also acquired. Students are given the opportunity of working on different computer and operating systems thus enhancing their resource network. The reports required of them are designed to teach the students how to create and implement their projects. The reports are also designed to develop the writing skills of the students. Advanced Problems in Physics and Engineering Physics I and II has met and exceeded our expectations by giving our students skills that will serve them in the future.

Artificial Neural Networks for Undergraduates

Patrick J. C. Lamont
Department of Computer Science
Western Illinois University
Macomb IL 61455
309-298-1452
mfpjl@uxa.ecn.bgu.edu

Abstract

The successful commercialization of neural net models has given real-world value to their inclusion in undergraduate artificial intelligence classes. Interest in neural nets as simplified human brains, parallel distributed systems and connectionist architectures is omnipresent. Students want answers to many questions. What are the characteristics of neural network models? When are neural nets used? How does one start neural network development? What are successful commercial applications? Since tools are available for neural network development and since much literature on example problems exists, neural nets are tractable and cost justifiable for undergraduates. Here we present basic ideas needed for studying neural nets.

Introduction

An artificial neural net, a system with architecture inspired by the human brain, consists of levels of interconnected processing units called neurons or neurodes or nodes capable of self organizing when trained with example data. Ideally neural nets are used for problems for which an algorithmic solution is believed to exist but remains unknown or is too complex. Self organization permits the neural net to predict, classify and recognize patterns. Other application areas are modelling, noise filtering, optimization, control and encryption. Many such problems involve prediction of the value of one variable based on the values of certain other variables. Forecasting the Dow Jones Industrial Average would be a feasible application of neural nets. Neural nets can be implemented as chips or simulated in software. The beginning student may use the software associated with one of the standard texts, while the more advanced student may be ready for a shell or tool or simulator such as LNKnet to work with neural nets.

Neural networks do not obey the basics of von Neumann architecture. Neural nets are analog (continuous valued) and parallel. They do not contain instructions and data stored together. They store data in the form of patterns of weights, states and interconnections of the neurons. They respond globally to inputs and do not follow a fetch, execute and store cycle.

Ockham's razor[1], also known as the principle of parsimony, provides a strategy for making neural net simulations possible. The principle states that one should not multiply entities beyond necessity. More generally, the principle is that a theory should not propose the existence of anything more than may be needed for its explanation. The complexity, including the number of hidden levels of neurodes, in a neural net model should be kept to a minimum.

In the late nineteen forties, McCulloch and Pitts were responsible for early mathematical models of neurons. In the late fifties, Rosenblatt developed a neural network capable of solving only linearly separable problems (i.e. the outputs can be separated with a line). This type of net did not involve hidden layers of neurodes and was called a perceptron. Minsky and Papert's text on Perceptrons appeared in 1969. Subsequently, Rumelhart, Werbos, LeCun and Parker all independently discovered back-propagation, a method for training nets with hidden layers to solve nonlinearly separable problems. More recently the Boltzmann machine has been developed and used to solve nonlinearly separable problems.

Characteristics of Neural Nets

The human brain contains billions (approximately 1011) neurons and trillions of synapses. There are approximately ten thousand synapses per neuron in the brain. A synapse is the specialized junction between two neurons, present in the nervous system of all animals. Nerve impulses at the axon (interconnection) terminals of one neuron are transmitted chemically and/or electrically to influence the excitability (activity) of the other neuron. Dendrites project from the cell body and receive information from other neurons. Signals coming through synapses on the dendrites excite, while those coming through synapses on the cell body inhibit. The neuron is responsible for adding excitatory signals and subtracting inhibiting signals. The result is compared to the threshold of activation of the neuron. If the result is higher, a signal is sent to other neurons over the axons.

Mathematically a network is represented by a graph which by definition consists of nodes and connecting arcs. Biological brains are highly complex, fault tolerant and parallel. To emulate the brain simple models of neurons are used as nodes in a network of input and output levels along with hidden layers of intervening nodes. VLSI neural network chips are custom designed for various applications. Alternatively, a processor board or computer can be used with a neurode or group of neurodes assigned to each processor. The VLSI implementations are extremely fast and compact.

In an artificial neural net, each neurode receives a large number of input signals, causing the neurode to reach some level of activity. For high enough activity level, the neurode generates an output signal. Weights are associated with each connection to a neurode. Weights determine the numerical importance of each incoming signal to a neurode. Outputs are given by the activation of the transfer function of the weighted or net sum of the inputs. The activation function is generally a nonlinear transformation function and is used to flatten the net sum. A so-called sigmoid function may be used. The inverse of one plus the exponential of the negative defines the sigmoid function frequently applied to the net sum. An engineering problem may require hundreds or thousands of nodes. To deal with the many nodes one vectorizes the inputs, outputs and activation function.

In developing a neural net, adequate time must be set aside for its training. Suitable training data must be collected and the learning method chosen. The three principal types of learning algorithm are unsupervised, supervised and reinforcement learning. In unsupervised learning, training data is grouped according to metric properties. For supervised learning, back-propagation is currently widely used. Weights are adjusted to minimize the difference between calculated and desired outputs. Training halts when no further improvements are observed. Overtraining can lead to weak generalization. In reinforcement learning, the network is graded continuously on its performance during training.

Back-propagation computes an error function based on the difference between desired and actual outputs. Weights are adjusted to minimize the error function. The method is based on the differential calculus. It is slow and may involve identification of purely local minima.

Commercial Applications

Although road forks and intersections are still problems, ALVINN (the Autonomous Land Vehicle in a Neural Network) steers and navigates almost as well as a human driver. Inputs involve a camera and a laser range finder. ALVINN was trained using computer generated images and is implemented in a van with on-board computer. A human operator is ready to take control if ALVINN has difficulties.

Shea and Liu have reported on operational experience with a neural network in the detection of explosives in checked airline luggage. Luggage on a conveyor belt is bombarded with neutrons. Characteristic radiations are produced by different elements when so bombarded. A neural net is used to recognize the patterns produced by, for example, plastic explosive. A 98% success rate with only 7.8% false identifications was reported. The FAA has deemed the net to be software.

Hancock has described a doppler radar processing neural network experiment. In a military situation various types of vehicle can be distinguished from one another. Problems of ship recognition, previously successfully tackled using frame based expert systems, could also be dealt with using neural nets.

LeCun has reported on handwritten digit recognition with a back-propagation network involving 91% success with a 1% error rate. The system involved 256 input nodes, 4000 hidden nodes and 10 output nodes. The system can be used for zip code recognition. AT&T Bell Labs have devoted considerable time to this project.

Perceptrons have been applied to the problem of site location in DNA sequences. Sites are located at the beginning of genes. An 80% accuracy rate has been reported. Given an amino acid and its surrounding medium, the amino acid folds into a helix, sheet or coil structure. Quian and Sejnowski applied neural nets to the problem of determining which structure would appear. Their work on predicting the structure of globular proteins using neural network models tackles a problem involving very difficult x-ray crystallography.

Banks have used neural networks in various prediction capacities. Consider the problem of deciding whether a person is a good loan risk for the purposes of a first mortgage home loan. Information on sex (M or F), marital status (three categories), age (four categories), number of years of unemployment (five categories), income (four categories), and health (three categories) is collected by means of an easy form. A 21 bit input string would give an output of whether the person was a good risk or not. Training data is selected from the bank's previous records. The neural net would be used to advise the bank on making loans.

Barr and Mani report that in the investment management domain the more task-specific the neural net is, the more effective it is. Predicting changes in the Standard & Poor's 500 Index is reported currently to give good results five to ten trading days in advance. Sensitivity analysis, determining percentage change in the output caused by change in an input, was used to select indicators. The 21 indicators chosen involve prices from other markets, and indicators based on price, volume and put-call ratio. A total of 126 inputs involving historical or telescoped data from previous periods, 14 hidden nodes, and one output node were used. Sufficient training data is available. The use of multiple nets, trained with the same data, but using different parameters, and averaging the outputs was reported to work well in practice. Barr and Mani report that neural nets appear to hold great promise for investment management in the domain of forecasting.

Many applications have been publicized, including fingerprint recognition, speech synthesis, commodity trading, hand writing recognition, and credit card fraud. Few applications of neural nets are fully commercialized due partly to the fact that recognition sciences have only recently begun to reach market acceptance and partly to the fact that recognition applications do not achieve 100% performance. Initial research and development expenses, if not financed in academic institutions, make the commercial cost justification difficult for small software houses and private companies. The problems associated with providing good ergonomics including satisfactory user interfaces also must be successfully overcome.

Network Development

In order to be in a position to mass market neural network solutions to cognition and other problems, one must know when to use neural nets, how to develop neural nets successfully, how to commercialize neural nets, and have suitable example projects at the prototype research and development stage.

Neural nets are useful for problems involving variability and uncertainty such as prediction and classification problems. Prediction problems like stock market performance and transportation overbooking use past history to guess future performance. Handwriting and speech recognition and fraud detection are classification problems in that certain patterns have to be classified into known types. In the examples mentioned in the previous sentence large amounts of training data are readily available.

To develop neural nets successfully one must be sure to analyze the problem chosen in order to understand its true nature. One must try to recognize simpler cases of the problem and ask if they have already been solved and by what method. Thus in order to tackle the cursive hand writing problem using neural nets, it is important to know or at least study solutions of the digit recognition problem using neural nets. Many interesting problems will present themselves. The distinction between real time and scanned input to the cursive handwriting problem must be studied from the point of view of which will lead to easier recognition by the automatic system. Calculus can be used to recognize maxima and minima. How can it be used to characterize written letters?

Problem definition can be reviewed many times to make sure that one has formulated a solvable problem. Data collection and preprocessing must be accurate and reflect a random sample of the data with which the system will cope. To design the network, Ockham's razor and/or the KISS principle must be remembered. A training method must be chosen and preparations made for user demonstration. Field testing of the neural net will lead to further refinements and possibly also redefinition of the original problem. Both commercial and public domain neural net simulators are available as tools. The commercial systems have the strength of technical support by the manufacturer, while a public domain system has the strengths of keeping costs low and a large user base. Difficulties associated with the size of the network, insufficient training data, the wrong choice of architecture or training method, and over training the network may lead to the false identification of a problem as intractable. A simple change such as replacing floating point arithmetic by integer arithmetic may make the problem appropriate with the available technology.

Linear programming problems can be solved using neural networks. However, when an algorithm is known for part of the solution to a problem, it is perhaps better used in conjunction with neural net methodology. Neural nets can also be used as tools for identifying rules for knowledge based systems. The rules might be difficult to identify otherwise, for example in the financial forecasting world. Neural nets can thus be used as a building block, as well as an alternative method, for solving problems.

A Recognizer

The cursive handwriting problem has been studied by Rumelhart who has recently been involved with a neural net recognizer. With a vocabulary of 1000 words and a reasonably large group of writers, the correct word is top ranked 80 percent of the time and is in the top five about 95 percent of the time. Individual writers with good penmanship can adjust to the recognizer to ensure better results. With a vocabulary of 100 words near perfect performance is possible provided easily confused words are excluded from the vocabulary.

The words of Basic English as given by Ogden augmented with number names, digits, isolated characters and upper and lower case were recorded, randomized and presented acoustically to writers. A digitizer was used for data collection. Coordinates and position of the pen (three possibilities) were recorded every 10 milliseconds stopping recording if the pen was removed from the paper for 500 milliseconds. Each writer produced files named by the writer's initials and the word written. The number of writers involved was 100 and 1.5 hour sessions were employed. Considerable amounts of data were generated. Hollerbach's "coupled oscillator" model of the data led to substantial data reduction. In the model, x and y component velocities (x1 and y1) of the pen are described by amplitude times the sine and cosine of a function of frequency, phase and time. When y1 is zero we begin a new stroke. The strokes are used as input to the recognizer network. The model provides a good data fit as modified by Rumelhart.

Positional rather than temporal ordering was used along with a set of five computed feature parameters for strokes. The five parameters are the net motions in the x and y directions, the net motion halfway through the stroke, the velocity in the x direction at the end of a stroke, and the ratio of the frequency in the x direction to the frequency in the y direction. The fifth parameter is taken to have value 1 or 2, the latter being required for a capital "S" or an "8". Characters are scaled for size before input to the network.

The net is designed to simultaneously segment and recognize patterns. The net should determine probabilistic presence of a letter in a pattern, but the location of the letter is not required. Translation independence is important in order to back propagate non-location specific errors to train an underlying location specific network.

Input consists of feature values for the strokes of the word to be recognized. In the net described by Rumelhart, since the cursive letter m contains the maximum possible six strokes, each node looks at six input features. Hidden levels of equally weighted nodes have one level for each hidden feature type. The activation function value of each node measures the probability that a hidden feature is present in the corresponding input. Thirty levels of hidden units were used.

A row of detection nodes is associated with each character type with each node taking input from the hidden feature units in its immediate vicinity. To support translational independence, weight linking is used to transfer automatically anything we learn about features or characters at any location to every appropriate location. The network is completed by providing nodes to output the probability that at least one character of a given type is present anywhere in the input field. A suitable marketing strategy will ensure the commercial success of the unit.

Simulators

BehavHeuristics, Inc., 335 Paint Branch Drive, College Park, MD 20742, offer neural networks for forecasting and optimization. Thinking Software, Inc., 46-16 65th Place - Dept AIX201, Woodside, N.Y. 11377 offers a system to build your own neural net plus a working neural net that you can train. NIBS, Inc., 62 Fowlie Road, Singapore 1542 offers NeuroForecaster/GA 3.0 claimed to combine the power of neural net and genetic algorithms and to support a Genetica option that creates, evolves and optimizes network structures and control parameters. Ward Systems Group, Inc., Executive Park West, 5 Hillcrest Drive, Frederick, MD 21702 markets NeuroShell 2 claimed to be the most powerful and flexible tool available in the financial area. Maureen Caudill, the well-known author, commends NeuroShell 2 as a "fabulous product". Software Frontiers Systems, Inc., P.O. Box 8524, Mesa, AZ 85214, offers neural network development software for serious applications and claims back propagation, self-organizing nets, and included C source code.

Public domain simulators include MUME for MS-DOS 5.0 (Contact Marwan Jabri at marwan@sedal.su.oz.au), PlaNet anonymous ftp boulder.colorado.edu (128.138.240.1) in the directory "pub/generic-sources" with file name "PlaNet.5.6.tar.Z", and NeuralShell anonymous ftp quanta.eng.ohio-state.edu (128.146.35.1) in the directory "pub/NeuralShell" with file name "NeuralShell.tar". The address for Aspirin Migraines is Russel Leighton, Mitre Signal Processing Lab, 7525 Colshire Drive, McLean, VA 22101 (russ@dash.mitre.org).

Footnotes

1. William of Ockham (c. 1285 - c. 1349) was a Franciscan from Surrey, England, who during his life had to answer heresy charges at the time of Pope John XXII in Avignon and is believed to have died of the Black Plague in Munich.

References

1. Bisant, D. and Rumelhart, D. E., Neural Networks: Introduction, Developments, and Commercial Applications, Western Institute in Computer Science (August, 1993).

2. Caudill, M. and Butler, C., Naturally Intelligent Systems, The MIT Press (1990).

3. Handcock, M., Doppler Radar Processing Neural Network Experiment, (1989). CSI Inc., 1280 Clearmont Street NE, Palm Bay, Florida 32905.

4. Hollerbach, J., An oscillation theory of handwriting, Biological Cybernetics 39 (1981), pp. 139 - 156.

5. Barr, D. S. and Mani, G., Using neural nets to manage investments, AI Expert 9, 2 (1994), pp. 16 - 21.

6. Kanade, T., Reed, M. L., and Weiss, L. E., New technologies and applications in robotics, Communications of the ACM, 37, 3 (1994), pp. 58 - 67.

7. McClelland, J. L. and Rumelhart, D. E., Explorations in Parallel Distributed Processing, The MIT Press (1988).

8. Nelson, M. McC. and Illingworth, W. T., A Practical Guide to Neural Nets, Addison-Wesley Publishing Company (1991).

9. Ogden, C. K., Basic English: International Second Language, (1968).

10. Quian and Sejnowski, Predicting the Secondary Structure of Globular Proteins Using Neural Network Models, Journal of Molecular Biology 202 (1988), pp. 865 - 884.

11. Rumelhart, Hinton and Williams, Learning Internal Representations by Error Propagation in Rumelhart and McClelland, (editors) Parallel Distributed Processing, The MIT Press (1986).

12. Zurada, J. M., Introduction to Artificial Neural Systems, West Publishing Company (1992).

Tale of Composition Wizardry: A Teacher and His Student Relate Their Adventures in Virtual Reality

John LaRoe, Matt Roberds
820 East 41st
University of Missouri - Kansas City
Kansas City MO 64110
rjlaroe@vax1.umkc.edu
mroberds@vax1.umkc.edu

The whole business of witchcraft is really centered around the magic of words.
-- N. Scott Momaday, [1]

This is the magic John found.

The first time I logged into InterNet, the first thing I earned to use there was IRC -- InterNet Relay Chat. I was filling time in the wee, wee hours, and I discovered that any hour of the day or night, I could find someone to talk to on IRC. A huge number of the people I met there were students, and of these most had logged into IRC in order to procrastinate on -- or maybe just take a break from -- their homework – especially writing assignments.

Ambushed by a renegade composition GTA, some of these last did not get far. They'd wind up talking to me about their essays, and neither of us would get much of a break.

But over time it dawned on me that these people were dodging out on their writing assignments by writing more fervently and furiously to each other about stuff they really care about -- sex, rock'n'roll, identity -- than they will ever write "to me" or "for me."

They hate writing. But they think this is fun. James K. Scarborough, a Master's degree candidate at California Polytechnic State University - Pomona, is doing some pretty exciting research that seems to confirm this, and to suggest still more. But I didn't know that at the time. I was just going with my instincts. I quit playing truant officer. indeed, I quit logging into IRC altogether.

But I started recommending that my students get mainframe accounts (@vax1.umkc.edu) and suggesting that they check out the e-mail, news, and, of course, IRC functions to be found there. I'll never be able to prove it, but I'd like to think some of the improvement I never saw in their essays might have shown up elsewhere.

Then I started getting e-mail from my students. Some of it was/is/always will be "why I wasn't in class today." (I delete those pretty much automatically, by the way.) Some of it has been "well, you told us to try out e-mail and yours is the only address I know." No harm in that, I suppose.

But a good chunk of it, too, has been questions, comments, and suggestions following on class discussions and writing assignments. And a good chunk of that has come from students who might not have been saying that much in the RL (real life) classroom.

Ken Wilson, a student of mine who happened to work in Academic Computing Services, suggested, I start using UseNet newsgroups," requiring my students to post weekly to an unmediated, but dedicated, locally hosted (again, @vax1.umkc.edu) computer bulletin board in lieu of the more frequently required personally maintained journals. I took him up on the suggestion and found the discussions livelier and marginally more thoughtful than those which had occurred in journal assignments I've used.

Ideas expressed and considered in one RL classroom discussion found their way into the newsgroup, where another class, reading the same novel and discussing similar issues, could respond as well. By the end of the semester, the newsgroup had attracted several readers (and a couple of contributors) from outside either RL class. Students were achieving RL publication in a VR environment. I liked it.

And it was right about then that Gary McDonald began his e-mail campaign. He was determined, I think, to show me that something very important was going on here (or somewhere in cyberspace).

Gary's our Vax Guy, senior systems analyst in Academic Computing Services. He started e-mailing me stuff he thought I'd get a kick out of, information on gopher, information on ftp sites, information about on-line e-zines, telnet addresses for various MU*s (about which I understood nothing) – forwarded e-mail, in other words, on assorted subjects arcane and virtual. Depending on how busy I was at any given time, it tended to pile up -- skimmed but not really read -- in my mail directory. I eventually had to apply for an expanded disk quota on Vax1 to accommodate my mail directory.

But scanning what had come to be an again overloaded MAIL directory one day -- when I knew I wasn't going to get anything done in RL, no matter how much there was to be done there -- I saw this one forwarded text that looked kind of interesting. Heck, it was down-right fascinating. I still don't know why I hadn't paid more attention to it when it first came in. It described a MU* named TinyTIM -- "not just a game, a really really big game."

I tried it. The interface was imperfect. The lines kept kind of floating across the screen. But, other than that, I liked it.

It was playful. It was anarchistic. It was fun. And it seemed to accommodate as much "chat" as IRC, only the chat here was contextualized.

If you're not familiar with MU*s -- they are mainframe-hosted, multi-user variations on the old CPM-based "Dungeons & Dragons" type games. Users 'move' from 'room' to 'room' using 'exits' listed in the room descriptions, and they may exchange messages with other users in the MU* -- you could think of them as chat programs with backdrops but there's a little more to them than that. Users may interact with the MU* environment too. (MU*s are more easily understood through exploration than explanation, and a wide assortment of MU*s are accessible via the mother of all gophers at UMN -- University of Minnesota, see "Fun & games.")

And what a backdrop! What a context! This was a virtual environment constructed by its users. It was plagued with typographical errors, not just in the discourse, but in the MU* itself. But that was because the MU* itself emerged from the users' collective and individual imaginations. It was, in two words, "a hoot."

I visited the Church of Tim, where a priest of Tim heard my confession and forgave my sins. I visited the Pop-Tart[tm] Church and toasted a pop-tart in the ceremonial toaster. I went to the belfry and rang the bell, with Grandpa Munster's help. A scantily clad waitress served me drinks in a Jamaican bar. Willem DaFoe took me on a personally guided tour of hell. I hung out in the Nexus and talked with Sketch, a guy cow and extremely talented wizard. I flew to Paris and toured the Grand Opera House (in which key scenes from "The Phantom of the Opera" played out before me as I wandered an underground maze) and La Theatre des Vampyres from the popular novels of Anne Rice. Then I walked down the street to London (no need for an under-the-channel tunnel in virtual reality) to visit the Mother House of the Malacascar, drawn from Rice's Tale of the Body-Thief.

It was maybe three hours later -- long after I'd turned my computer off and trundled off to bed -- that I went ballistic.

Scenes from Anne Rice Novels! Why You Could Do Any Novel, Huh? I could do any novel!

My first thought, not a good one, was to spend enough time in tinyTIM to earn enough timBUCKS to build a scene from _Ceremony_, which I was teaching that semester in two Eng 225 classes.

You see, users in TinyTIM (and several other MU*s, for that matter) earn money, which is of course required for building within the MU*, by simply wandering around. You'll enter a room and, along with the room description, see the message: "You found a penny!" I'm sure you can imagine the sort of positive reinforcement that provides an under-paid university lecturer. But I would have had to spend several hours there each day over the course of the semester to collect enough pennies to virtually reconstruct much. And by then, of course, it would be too late for this semester. And this really couldn't wait, so ...

... my next thought was, "maybe I can talk some of my students into helping, we could pool our pennies." I liked that idea better. But then I figured I would just make them do the whole thing, but give them all my pennies to speed the necessary accumulation of capital. I liked this idea best of all because it would give me a great excuse to hang out in TIM, but I wouldn't have to learn how to program stuff, I thought. (Needless to say, it hasn't worked out that way.)

I managed to doze off for about two hours, I think.

I talked to several of my students the next morning. They seemed game to try it. I talked to Gary about my plan and he thought I was really going about things the hard way. He said, You know, if it has an educational value, you could probably build your own right here on Vax.

Ken Wilson, a student in one of those 225 classes who works for Gary, then informed me that he and several other computer services employees had acquired a program for MUD-building a year or so previously.

I offered MUSH construction of or related to Leslie Silko's Ceremony as an alternative to one of three essays required in the class, at my discretion. Two students from one class, one from the other, and a fourth who was not enrolled in any of my classes responded: Ken Wilson and Matt Roberds from one class, Jeremie Day from the other, and Chris Bjuland from Computer Services. We met in the Vax room and at a pizza parlor. We worked out, between the five of us, a general framework for the MUSH:

We would use Kansas City landmarks (and my own fantasy Keyboard Cafe) as gateways to the literary worlds we would reconstruct. Enamoured of the free-style play of creativity I found on tinyTIM, I encouraged those students to recruit others and seek to charter a student domain within the MUSH. That's still under discussion.

We were building within the week.

Ken and Chris began work on a virtual interpretation of Sherman Alexie's short story, "The Trial of Thomas Builds-the-Fire" from The Lone Ranger and Tonto Fistfight in Heaven. I think I let that design get out of hand and it remains unfinished. But who knows? Someday? What's been started in that project is very entertaining. I want to see it completed and restored in an expanded MUCK, but more on that later.

The things that did get finished that semester were the gateway -- in which all of us have had a hand -- and an end-game, you might say, for Silko's Ceremony, constructed by Matt and Jeremie. It points, I think, to the spirit of Silko's message.

Started that semester and completed this semester was Mu Bedford Seaport (ca 1850) -- with portions constructed by myself, Matt, Craig Bartholomaus (Associate Professor for American Literature), and, of course, Herman Melville (without whose collaboration we would have been lost). Mu Bedford treats the adventures Ishmael had, and might have had, en route to signing on to The Pequod.

Under construction this semester (in a litMUSE2 established to provide a short-term, but stable construction environment while we seek out a new program and hardware) is a Kansas City casino (ca 1935).

Starting this second MUSE was made necessary because the original litMUSE is "full." Built in a very basic VMS-hosted program (Monsters, about which you'll hear more from Matt at the conference) we maxxed it out over the course of a semester and a half.

Running out of MU* capacity at mid-semester presented problems, of course, and it generated a certain amount of panic. But it also serves, I think, as a powerful validation of the notion: Students will write in these things. That's how it got maxxed out, well, that, and the stuff that I wrote.

The casino, at any rate, with adjacent bar and dance club, will provide a context for Harper Barnes' Blue Monday, a piece of historical fiction focused on Kansas City's Jazz age. It's being virtually constructed by Christian Daou, Pauly Gillespie, Audra Johnston, and Dean Stauffer -- all students of my current 225 class, which is reading Blue Monday.

And we're certainly not going to stop here.

Matt has laid a foundation for Berdellac's Castle, which when raised will host Sir Gawaine -- and test the young knight's manners -- on his quest to acquit his promise to the Green Knight.

Another composition teacher has expressed a sincere desire, I think, to virtually reconstruct two concluding scenes for The Natural, contrasting Bernard Malamud's with the one provided by Tri-Star Pictures.

I know a couple of literature professors who are intrigued with the idea of not having to cart their cardboard reconstructions of the Globe Theatre to class anymore. Virtual reconstructions can be toured through any phone line. (Actually, there's a lot we could do with the Globe. I mean, I work summers as a stage-hand, and I've got a few ideas on how we could improve the sound and lights in there too. You know, so they could book some more contemporary acts. But I haven't discussed these with the good professors yet.)

All this is lots of fun. And really, I see no end to the possibilities here for fanciful and literary MUSE-ing, but I see a whole lot more potential in MU* programming than that, potential that encompasses our campuses entirely.

Me, I'm seeing RooMU or UMKCmuck. You may see something different. But I'm suggesting we all start thinking "virtual campuses," here. Places ...

... where English professors lead their classes through the private chambers of Queen Elizabeth as envisioned by, say, Patricia Fumerton[2].

... where History professors take their classes through the Battle of Hastings step by step, with plenty of what-ifs along the way.

... where Biology professors take their students through micro-organisms, playing the parts of various anti-bodies.

... where Business and Economics professors and their classes sit in on corporate board meetings or meetings of the Federal Reserve Board of Governors.

... where Professors of Education situate their classes pretty much anyplace they can imagine.

... where law students argue the most important cases of our times, and future times, before a virtual Supreme Court.

Yeah, I know. these are all imaginary trips. But that's the power of them, and the beauty. Anybody out there remember radio theater?

I'd also like to suggest that a virtual campus would provide an environment in which many students might find side trips into other disciplines informative and rather more entertaining and accessible than anything we're apt to accomplish on a traditional field trip to the library.

Jeremie Day, mentioned above, writes that it also seems to her "that one virtual world could serve as a resource for more than one type of class. For example, a historically based world could serve as a reference for any type of class from natural history and history to literature and the arts" [3].

I'm not the only person deliberately exposing undergraduates to MU*s, and doubt very much that I'm the only person around who's looking beyond the cyber-classroom to the virtual campus. I'm just the guy who wrote this.

One other educational MU*s I know of geared to specific undergraduate use is:

BioMOO

Telnet: bioinformatics.weizmann.ac.il 8888

David McKalip, M.D., Division of Neurosurgery, UNC-Chapel Hill

dmmckali@gibbs.oit.unc.edu

919-966-1374

Described, by McKalip, as "a very active social virtual reality biologic scientific community."

another is:

WriteMUSH

Telnet: palmer.sacc.colostate.edu 6250

A MU* seemingly devoted entirely to composition instruction.

I know there are others. I am told, for instance, that a MUSE currently under construction at the University of Maryland - Baltimore will be used in the study of Greek mythology. I have no idea how many more there may be. I suspect several -- perhaps enough of us that the virtual campus is no longer up for discussion in terms of should we or shouldn't we. We're going to. After we've been doing it a while, we can look back and say this worked really well or that had this other unexpected result. Some of those unexpected results, we may not like. We'll have to see.

The current objective, it seems to me, is to master the conventions of this new medium while searching out as yet unimagined possibilities and watching out for unexpected problems.

End Notes

[1] qtd. in Woodard, Charles L. Ancestral Voice: Conversations with N. Scott Momaday. Lincoln, NE: U of Nebraska P, 1989.

[2] Patricia Fumerton: in her essay "'Secret' Arts: Elizabethan Miniatures and Sonnets." Representing the English Renaissance. Stephen Greenblatt, ed. Berkeley, CA: U of California Press, 1988. mentioned just because I like her essay and because I think the so-called new-historical approach is especially well-suited to expression and exploration in the MU* environment, but that's probably a whole 'nuther paper, for a writer with a whole different set of credentials. Rats. --LaRoe

[3] Day, Jeremie. "novelette re: MU*." e-mail of 20 April 1994 to RJLaROE.

Tale of Composition Wizardry
a student relates his adventures in Virtual Reality

Matt Roberds

"You don't have anything, if you don't have the stories." (Silko 2)

This is Matt's story.

I was a student in John's English 225 (English Composition II, the sophomore-level class at UMKC) during the fall semester, 1993. A few weeks into the semester, John started encouraging us to get accounts on UMKC's VAX, if we didn't have them already, and to learn how to use the email facilities on the VAX. He got excited about the tremendous variety of information available on the Internet, and was trying to show us how we could make use of that in researching our papers for his class. Soon, he had arranged for a local Usenet newsgroup as an alternative forum for class discussion. We spent a class period learning how to log in, send and receive mail, read and post to News, etc.

But the involvement with computers didn't stop there. A little later on in the semester, John came to myself and Ken with an idea. He wanted to somehow turn the book we were reading in class, Leslie Marmon Silko's Ceremony, into an interactive environment on the VAX. Ken had already had some experience with a program for the VAX known as Monster. Monster is sort of a primitive MUSH/MUD/MUCK/MOO that only runs, as far as I know, on VMS systems. Anyway, it gives you a basic framework of rooms, exits, and objects, which you can describe, customize, and interconnect in a myriad of ways. Ken thought that this would be a good starting place in our attempt to transform a text into a virtual environment. John also had a student from his other section of English 225, Jeremie, who was interested. Chris, who works for Academic Computing Services with Ken, also decided to join us.

After we got approval (much to our amazement) from the system administrators, Ken got Monster loaded and compiled on VAX. We dubbed it "LitMUSH", for "Literary Multi-User Shared Hallucination." Ken then proceeded to show Jeremie, John, and myself how it worked. After a few minutes of really dumb questions, Ken instead resolved to write a manual for novice builders. His manual has proved quite helpful - both to us, and to John's students this semester. After reading the manual and messing around in LitMUSH, building and linking exploratory and "test" rooms, we all felt that we knew just enough to really get ourselves in trouble.

Then came the slight problem of "who's going to do that?" We hadn't really discussed the division of labor on the project, which led to some misunderstandings in the early stages. We finally all got together and discussed exactly how we were going to turn the scenes in the book into rooms and exits in Monster. This included things like:

How much of the book do we want to include?

For Ceremony, Jeremie, John, and I decided that we would work on the conclusion of the novel. Near the end of book, the main character, Tayo, is presented with a choice as to killing another character, Emo. In the book, Tayo does not kill Emo. As part of the "get 'em to read the book" thinking, we wanted to also include an option for Tayo killing Emo. This way, users could choose the "wrong" ending, still play for a little while, then get scolded for making the wrong choice (perhaps by placing their character in a room with no exits, or by making them replay the game until they get it right.)

What rooms do we need? How should they be connected together?

This involved lots of thinking. Does this scene really mean anything, or can we gloss over it? What things must we include? What choices should we give the user that they didn't get in the original text, and what should happen if they pick one of these new choices? These kinds of questions really force you to look at the book again and try to see the plot and character development more clearly than you did the first time. I must have read the book at least 2 or 3 times, noticing different ideas and different applications to LitMUSH each time. I know Jeremie did a lot of study of the book also. We finally hammered out a map, on paper, of the different rooms and the links between them.

Who should do what?

We had a map of the rooms and exits we needed, but the map wasn't going to type in all the descriptions for us. We had to decide who was going to be responsible for typing this room -- that exit description -- creating those objects? I think this is one of the critical things, as it would be in any group endeavor...making sure everyone knows what they are supposed to be doing. This is where some misunderstandings arose at first, until we all got together and worked it out.

Once we had all of the above, we could have at. Transferring the text of the book to the VAX proved to be a challenge. Monster provides up to twenty lines per room of descriptive text, and ten lines of description for each exit. However, you can't use all 30 of those lines, because the screens most people will see it on only have 24 or 25 lines -- your masterful adaptation will simply scroll off the screen before the user gets a chance to read it! We had to find a balance, between terse and unsatisfying descriptions and pages and pages of minute detail. We also had to find a way to decide when to paraphrase the book and use your own words, and when to let the author's words stand. (To cover citation issues, each "book" in LitMUSE has a small sign or a note in one of the initial rooms, which when 'looked' at gives an MLA citation for the book used.) I think this whole process was much harder, and more educational, than the paper assignments it replaced. It almost forces you to take a harder look at a book, and think more about it, than you would in simply reading it for class.

Another interesting event was the Monster crash part-way through the work. We needed more space to build in, but found we couldn't just allocate more space "on the fly." Ken had to shut down the entire thing, and start again from scratch, this time adding more space at start-up. As I write this, late March, it is down again - we have run out of space in another file. This points out the need to know the limitations and upgrade possibilities of one's software (and hardware, although we have not had any major hardware problems) before undertaking a project like this. Anyway, we managed to salvage some of the text and room structure and transplant this to the new copy of Monster. Some of the salvaging was made easier by UMKC's Xwindows terminals - these made it possible to work on text in a word processor in one window, and then cut-and-paste it into the LitMUSH window. This process could be easily duplicated on a PC with Windows or on a Mac.

After many weeks of work, worry, revision, work, jubilation, despair, work, revision, and work, we decided to call it done. The fact that it was the end of the semester, and John needed to get his grade reports turned in, also was an influence on our decision. :) Apparently John liked what he saw. He had been busy working on a grant proposal, which got accepted by the Sosland Fund at UMKC. This provided funding for a faculty adviser (him) and a student assistant (me) to run the LitMUSE. (Literary Multi-User Shared Environment. The change from "hallucination" to "environment" was one, as they say, for marketing reasons.)

This semester, work continues on LitMUSE. John and I continue to edit areas that are already "completed", like Ceremony. Myself and John are also working on an adaptation of Melville's Moby Dick, and other instructors in the English department have shown interest in virtually re-creating works such as Sir Gawain and the Green Knight and Bernard Malamud's The Natural. John put together the Keyboard Cafe for new users, while I set up Union Station, a virtual reconstruction of Kansas City's real Union Station. It is used as the central point that links you to the various works in the MUSE. Three students from John's 225 class this semester are working on Harper Barnes' Blue Monday, set in Kansas City of the 1930s. We are also looking at the prospect of getting a machine devoted to LitMUSE, and running one of the more powerful MUD/MUSH/MUCK programs. We feel that it is a valuable tool for teaching composition, and that it could and should be successfully adapted by others.

"Take it back.
Call that story back.
...
It's already turned loose.
It's already coming.
It can't be called back." (Silko 38)

Works Cited

Silko, Leslie Marmon. _Ceremony_. New York: Penguin, 1986.

Internet Access, Navigation, and Resource Location: The Future is Here

Bill Z. Manaris, Christopher A. Jones
Department of Mathematics and Computer Science
Fayetteville State University
1200 Murchison Road
Fayetteville, NC 28301
910–486–1666
manaris@fsu.fsufay.edu
chris@fsu.fsufay.edu

Abstract

This paper presents our efforts in preparation for August 1994, when Fayetteville State University will be connected to the North Carolina Information Highway, the first state–wide fiber–optics network in the United States. Our intention is to provide our students and faculty, novice and experienced users alike, with faster, more intuitive, and more powerful means to access information via Internet than standard telnet and ftp. Specifically, we are in the process of locating, installing, testing and evaluating various Internet access, navigation, and resource locator tools. These tools include archie, gopher, lynx, mosaic, and USENET news readers. This work resembles a bootstrapping process where existing tools are being used in locating new and more powerful ones, which in turn are being downloaded and installed. This process is currently being performed across several platforms, including multimedia workstations, file servers, a Sequent/Dynix parallel computer, and a VAX/VMS cluster.

Introduction

Fayetteville State University (FSU) is one of the sixteen campuses of University of North Carolina system that will be connected to the North Carolina Information Highway (NCIH). NCIH is a state–wide fiber optics network backbone operating at 155Mbs that will facilitate efficient transfer of data, voice, images, and two–way video. In preparation for August 1994, when the NCIH will finally reach us, we are developing the necessary infrastructure for our end of the network. This infrastructure consists of conduits among major campus buildings, internal building wiring, fiber optic cable, network hubs (file servers in major campus buildings), network cards for faculty workstations and workstation labs, wiring closets, and necessary software.

Our presentation focuses on the software part of the process, as it is experienced within the Mathematics and Computer Science department. Our goal has been to locate, download, and install tools that would allow our computing community to access andnavigate through Internet, and locate needed information and resources. These tools include archie, gopher, various USENET news readers, and WAIS and WWW clients. The hardware platforms involved in this process include our departmental Sequent/Dynix parallel computer, multimedia and other workstations 36, and a VAX/VMS cluster.

Internet Access and Resources

Internet started approximately 22 years ago, under the name ARPAnet, as an experimental network for the military. ARPAnet's main characteristic was that it could continue to operate even during partial network outages [6]. Currently, Internet is a global network of computers with vast amounts and types of information and services available. Figure 1 identifies how to access detailed administrative and technical information on Internet. As this figure suggests, to access resources on the Internet requires (a) having computer expertise and familiarity with ftp (and possibly telnet), and (b) maintaining Internet addresses of important resources. For those that are inexperienced with ftp and telnet, these vast resources have been out of reach.

Navigation and Resource Locator Tools

A new generation of systems is now available on the Internet for navigation and resource location. These tools make gathering of information a keystroke or two away and can be installed on just about any system with Internet access. Such systems include archie, gopher, WAIS, and WWW. Herein we will concentrate on our experiences with locating, downloading, and installing gopher, lynx, mosaic, and the nn USENET news reader. For information on other such tools, the reader should consult [2, 3, 4, 6, 7, 9, 11].

Client/Server Systems

Usually, tools for navigation and resource location are implemented by two separate programs, namely a server and a client. The server program provides the information, whereas the client program collects the information and makes it available to users [4]. If the main objective is resource location, then only the client version of such tools needs to be installed on a given system. On the other hand, if one wishes to make resources available to the Internet community, then the server version of such tools needs to be installed.

Topic	*Method Access*	*Internet Address*	*Resource Name*
Internet Growth	ftp	tic.com	matrix/growth/internet/
Internet History	ftp	umcc.umich.edu	pub/seraphim/doc/nethist8.txt
Internet Maps (Europe)	ftp	eunet.fi	nic/pub/netinfo/maps/
Internet Maps (Many)	ftp	ftp.uu.net	inet/maps/
Internet Maps (NSFNET)	ftp	nic.merit.edu	maps/
Internet Maps (SuraNet)	ftp	ftp.sura.net	pub/maps/
Internet Size	ftp	ftp.nisc.sri.com	pub/zone/
Internet Statistics	ftp	nic.merit.edu	nsfnet/statistics/

Figure 1. How to Access Administrative and Technical Information on Internet [2]

Locating Resources with Archie

Since our goal is to provide the FSU computing community for access to available Internet resources, we have concentrated on installing client versions of navigation and resource location tools. To locate these tools, we downloaded and installed archie, a system which can locate resources on the Internet. Once archie was installed on our system, we were able to locate additional tools. For more information on archie, see [9]. Figure 2 shows a portion of the output generated by archie when asked to locate copies of itself. This information may be used by the reader to download and install archie on his/her system (if it is not already available).

The following sections identify some of tools we selected, along with highlights of the installation process on our system. As indicated above, these tools are gopher, lynx, mosaic, and the nn USENET news reader.

```
Host calypso-2.oit.unc.edu
  Location: /pub/packages/infosystems
    DIRECTORY drwxr-xr-x     1024 Mar  7 08:53  archie

Host ftp.gsfc.nasa.gov
  Location: /pub/info-servers
    DIRECTORY drwxr-xr-x      512  Sep 20 1992  archie

Host ftp.sura.net
  Location: /pub
    DIRECTORY drwxrwxr-x      512  Mar  3 1992  archie

Host nisc.jvnc.net
  Location: /pub/MSDOS
    DIRECTORY drwxrwxr-x      512  Sep 23 1992  archie

Host yuma.acns.colostate.edu
  Location: /software.ibmpc
    DIRECTORY drwxr-xr-x      512  Feb 20 1992  archie
```

Figure 2. Location of Archie Source Distribution

Gopher

Gopher is a system developed at the University of Minnesota that allows users to search and browse distributed information on theInternet. It was developed in April of 1991 by a team consisting of Bob Alberti, Farhad Anklesaria, Paul Lindler, Mark McCahill and Daniel Torrey [4]. This tool was designed originally so that departments at the University of Minnesota could cheaply and easily provide information campus-wide.

Gopher organizes available information in menus. Each menu consists of a sequence of numbered menu items. Figure 3 shows the opening menu of The News & Observer newspaper's gopher server [8]. Each menu item may be selected either by moving the cursor up and down with the arrow keys or by typing the corresponding number.

When an item is selected, gopher's action depends on the type of the selected item. For instance, an ASCII–file item will be fetched and displayed, a binary–file item will be downloaded, a directory item will be entered and displayed as a new menu, and a system–pointer item will initiate a telnet session to the specified system. In summary, gopher simplifies navigation and resource location by providing an abstraction barrier that hides the location of Internet resources and the means through which they may be accessed.

Basic gopher commands include arrow keys, i.e., j (previous menu), i (previous menu item), k (next menu item), and l (select current item), m (main menu), ? (help), q (quit), y (yes), n (no), and u (previous menu). Examples of gopher usage include downloading information on NSF grants, reading the current weather forecast for Charleston, SC, and accessing the on–line issue of The Chronicle of Higher Education [1].

Installation of Gopher Client

Our Sequent/Dynix version of gopher originated from boombox.micro.umn.edu (network address 134.84.132.2) in directory /pub/gopher. After downloading version 2.012 and decompressing it, we found that it was necessary to make minor changes to two installation setup files, namely config and makefile. Additionally, through archie, we were able to locate appropriate versions of gopher for our workstations and the VAX/VMS cluster.

```
          Internet Gopher Information Client 2.0 pl6

              The News & Observer (Raleigh, NC)

—> 1. About this Gopher service.
   2. About The News & Observer Publishing Co./
   3. Today's edition of The News & Observer (a sampler)/
   4. The Insider: North Carolina Government/
   5. Other sources of news, sports and weather/
   6. Education resources/
   7. Exploring the Internet/
   8. Government information/
   9. Just for fun/
   10. Kidslink/
   11. Libraries/
   12. MetroMUD: You're not in Raleigh anymore ... <TEL>
   13. Misc. Triangle–area resources/
   14. Test Section/
   15. The Armchair Traveler/
   16. The Bookshelf/
   17. The Electronic Reference Shelf/
   18. The Journalists' corner/

Press ? for Help, q to Quit, u to go up a menu      Page: 1/2
```

Figure 3. Gopher Menu Example [8]

World–Wide Web (Lynx and Mosaic)

The World Wide Web (WWW), also known as The Web, was developed by CERN, a European consortium of scientists based in Geneva, Switzerland [4]. WWW is an Internet–wide client/server system comprising of hypermedia information. Hypermedia files allow users to access different types of information, such as text, images, video, and sound. There is no main menu, or a central point, as in gopher. Instead users enter WWW at any location and may navigate around the world by following resource pointers.

Lynx and mosaic are two client systems offering access to WWW. Additionally, they offer access to finger, ftp, gopher, USENET, WAIS, Whois, X.500, and a variety of other services [3, 10]. In lynx, for example, you would get a page of hypertext information on your screen. On that page certain words are highlighted (see Figure 4). These highlighted words are hot–links, i.e., pointers to other WWW locations. These hot–links may be activated by placing the cursor on them and hitting the RETURN key. A hot–link activation will display a new page of information, which may be on a system which is physically located thousands of miles away from the system storing the previous page. For example, the activation of the hot–link Information Resource Meta–Index in Figure 4 will access/display the screen shown in Figure 5 [5]. Technically, hot–links are abstractions of uniform resource locations. A uniform resource location (URL) is a description of the type of the resource being accessed and its Internet location [4].

As indicated above, both lynx and mosaic can serve as WWW clients. The only difference is that mosaic can access all types of hypermedia information, such as text, image, sound, and video data, whereas lynx can display only hypertext information, i.e., text data. This means that lynx may only use highlighted words as hot–links, whereas mosaic allows various types of data to serve as hot–links, such as words and images.

Additionally, client systems are very useful for local information distribution, since one may set up hypermedia files, and make them available to local users. A WWW–server program isrequired only if this information is to be made available to other systems over the network. For example, on our Sequent system we have set up hypertext information on the project discussed herein, its objectives, and its current status. Additionally, a hot–link has been included to allow users to enter their comments via electronic mail.

FSU Lynx Client

WELCOME TO FSU LYNX AND THE WORLD OF THE WEB

You are using a WWW Product called Lynx. This system was installed by Chris Jones under the auspices of the NIR project at the FSU Math and Computer Science Department.

INFORMATION SOURCES ABOUT AND FOR WWW
- For a description of WWW choose Web Overview
- About the WWW Information Sharing project
- WWW Information By Subject
- WWW Information By Type

OTHER INFO SOURCES
- University of Kansas CWIS
- Nova–Links: Internet access made easy
- NCSA: Network Starting Points, Information Resource Meta–Index
- Hytelnet database: Telnet information resources
- All the Gopher servers in the world

Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<–' to go back

Figure 4. The Home Page of the FSU Lynx Client.

Installation of Lynx and Mosaic

Both lynx and mosaic are available on our campus. Specifically, lynx version 2.2 was downloaded from stat1.cc.ukans.edu (129.237.33.2) and has been installed on our Sequent/Dynix system. In contrast to the gopher client, this installation process was not straightforward. The main problem was that the lynx distribution did not support our platform, so we had to make major modifications. Fortunately, lynx's developer, Lou Montulli of the University of Kansas, assisted us in overcoming all major difficulties. In summary, we had to locate and install a WAIS client (WAIS is an Internet–wide indexed text–retrieval system [11]), modify the lynx source code, and customize the installation files.

Once we successfully installed lynx, we proceeded with developing our own home page. The home page is the hypertext page displayed when lynx is invoked without any parameters. Hypertext and hypermedia pages are defined using the HTML language. This language allows inclusion of textual information, in addition to definition of hot–links and corresponding URLs.

Mosaic was downloaded from sunsite.unc.edu (198.86.40.81) and was easily installed on Windows 3.1 workstations. This implementation of mosaic, namely WinMosaic, requires a TCP/IP connection and a special file to provide TCP/IP networking within MS–Windows, namely WINSOCK.DLL.

USENET News

```
                    Internet Resources Meta-Index (p1 of 6)
              INTERNET RESOURCES META-INDEX
   This document is intended to be a loosely categorized meta-index of
   the various resource directories and indices available on the
   Internet. Please send suggestions for new entries on this list (any
   index or directory of resources) to ebina@ncsa.uiuc.edu.
World Wide Web
   * Searchable indices:
      + W3Catalog, a searchable catalog of WWW resources at CUI,
        Geneva, Switzerland; updated automatically every day from a variety of sources.
      + WWWW - the World Wide Web Worm
      + Meta-Library index at the Global Network Academy.
   * Subject indices:
      + Whole Internet Catalog from O'Reilly and Associates, a part
       of the Global Network Navigator.
      + WWW Virtual Library, a subject listing from CERN.
   * Server indices:
      + Central index of WWW servers at CERN.
 — press space for next page —
```

Figure 5. Internet Resources Meta-Index [5].

USENET news started as an experiment in North Carolina in 1979. The general idea was to create an electronic bulletin board [4]. Initially the University of North Carolina and Duke University were only two USENET sites. The exponential growth of theInternet in the last two decades caused a proportional growth of available USENET news sites.

USENET allows people from around the world to discuss a large variety of topics [4]. These discussions are organized in several thousands of newsgroups in which a user may post or from which a user may retrieve articles on a particular subject. Each newsgroup is identified by a unique identifier which consists of several fields. The first field identifies the main subject category. One example may be comp (a newsgroup category related to Computer Science). The other fields identify subcategories. For example, comp.ai.nat-lang contains articles on Natural Language Processing. A user may subscribe or unsubscribe to any newsgroup (s)he wishes. Subscribing to a newsgroup means that a particular group of articles on a specific subject is made available to the user when reading news.

Accessing USENET

In order to access USENET news you must first have a newsreader program, which will allow you to read and possibly post articles. These articles are physically stored on a news server which may be the same computer as the one running the news reader, or some other computer reachable through the network. A newsreader allows you to subscribe and unsubscribe to newsgroups. The choice of newsreader and the speed of your connection to the news server will determine the speed of your USENET access. Most newsreaders employ an intermediate program that communicates with the news server. This is based on a standard protocol called Network News Transfer Protocol (NNTP).

Installation of nn News Reader

There are many different newsreaders available. Some of the more common systems include rn, trn, tin, and nn. Each program offers a unique interface to reading news. We decided to install the nn newsreader because of its ease of installation and particular aspects of its design that made it faster than other newsreaders we had tried. Our version was downloaded from ftp.uu.net (192.48.96.9).

One of the design aspects mentioned above is that nn maintains a local database of available articles consisting of header information, such as author name, subject, date, and size, and a pointer to the actual message on the news server. Thus, the disk space required for maintaining USENET access is kept minimal and, simultaneously, the user has almost instant access to a list of available articles. The nn user browses through the list of such articles (see Figure 6), and only when (s)he decides to access a specific article will it be downloaded from the news server. Nn uses a process called nnmaster which runs periodically in the background and compiles a list of available articles by connecting to the news server.

One of the problems we encountered was that our nnmaster was not updating the local news database regularly. This resulted in degradation of news access speed. After several unsuccessful experiments, we decided to use our installation of nn to post an article to the newsgroup news.software.nn describing our problem. Amazingly, we got immediate responses from two individuals, one from Australia and another from the US, who where experiencing similar problems. Eventually, the problem was identified as the slow connection to the news server. Due to the slow network access time, nnmaster would be unable to complete its collection task in the allotted time (10 minutes). Additionally, the news server would sometimes drop nnmaster's connection. Since our network connection will be upgraded in August 1994, this is a temporary problem. Our solution to this problem has been to reduce the frequency with which nnmaster connects to the news server from every 10 minutes to once every hour; additionally, nnmaster updates only certain newsgroups each time, as opposed to all available newsgroups.

Conclusion

Fayetteville State University is preparing for the future by building the infrastructure which will provide access to and utilization of the North Carolina Information Highway. Currently, in addition to the hardware installation in progress, we are experimenting with various Internet navigators and resource locators. Through these tools:

- an unimaginable wealth of resources is now available to non–expert computer users,
- local information and services, such as library access, student and faculty information, and general campus news, may now be exchanged with insignificant overhead and expense,
- local accomplishments and news may be easily distributed to the outside world, and
- our students, faculty, and general community may efficiently stay informed, and even participate in defining the–state–of–the–art in any field of interest.

The work reported herein is only the beginning of what we are planning to accomplish at FSU. Future goals include installing servers for gopher and WWW on our campus. This will allow the rest of the Internet community to access our systems and obtain local information. Additionally, we are continuously searching for available tools and resources so that we may bring the latest to our computing community. If nothing else, we have given our students and faculty the opportunity to explore the world and experience for themselves the wealth of knowledge available via Internet. We hope that, through the installed tools and services, the terabytes of information that will be readily available through the NCIH, will have more meaning to ourstudents and a major impact on their academic development.

```
   Newsgroup: comp.ai                      Articles: 34 of 255/8

a  Erik Max Francis     17  >NPC planning
b  Ben Bongalon         28  >>>>
c  Steven Grady         34  >>>>>
d  Jorn Barger         110  >>AI, Ethics, and the Schank Syndrome
e  Jeff Bishop         120  >>>
f  Andrea Chen          37  >>Turing Test
g  Spider Little         9  Bayes Networks Question
h  bob dick             24  >Perceptual Control Theory
i  haroud@liasun6       69  Workshop and Seminar on Applied Constraint Reasoning
j  Thomas Wolf          29  references: graph isomorphism
k  Thomas Wolf          21  case representation (CASUEL)
l  W J Rapaport         59  >Directions to another newsgroup ?
m  John Galletly        45  Research Studentship
n  Tom Gordon           79  ANNOUNCE — The qwertz Toolbox
o  Hendler Jim         232  CFP ISATA 1994
p  Y Cengeloglu        131  Blackboard, Dynamic <> Agents tool. (DYNACLIPS V3.0)
q  jones@ils.nwu       203  New MA in Learning Sciences
r  Youngwhan Lee        21  >Passing functions/conditions as parameters
s  jmcglothlin          14  >Expert System
   — 16:00 — SELECT — help:? ——Top 54%——
```

Figure 6. Using nn to read the newsgroup comp.ai.

Acknowledgments

The authors would like to thank the following individuals: Eric Harper and Mark Rundlett of the FSU MIS department for their technical support and assistance; Lou Montulli of the University of Kansas, for his assistance with lynx installation; Andrew Disseldorp of the University of Melbourne, Australia for his assistance with nn fine-tuning.

References

1. Academe this Week (The Chronicle of Higher Education). Available via gopher from chronicle.merit.edu port 70.

2. December, J. Information Sources: the Internet and Computer–Mediated Communication. Release 3.15, December 13, 1993. Available via ftp from ftp.rpi.edu in file pub/communications/internet–cmc.txt or via lynx/mosaic at URL http://www.rpi.edu/Internet/Guides/decemj/internet–cmc.html.

3. December, J. Internet Tools Summary. Release 1.43, March 24, 1994. Available via ftp from ftp.rpi.edu in file pub/communications/internet–tools.txt or via lynx/mosaic at URL http://www.rpi.edu/Internet/Guides/decemj/internet–tools.html.

4. Hahn, H., and Sout, R. The Internet Complete Reference. McGraw–Hill, 1994.

5. Information Resource Meta–Index. Available via lynx/mosaic at URL http://www.ncsa.uiuc.edu/ SDG/Software/Mosaic/MetaIndex.html.

6. Krol, E. The Whole Internet û User's Guide and Catalogue. O'Reilly & Associates, Inc. 1992.

7. LaQuey, T., and Ryer, J.C. The Internet Companion Plus û A Beginner's Start–up Kit for Global Networking. Addison–Wesley Publishing Co., 1993.

8. The News & Observer (Raleigh, NC). Available via gopher from merlin.nando.net port 70.

9. Obraczka, K., Danzig, P.B., and Li S.H. Internet Resource Discovery Services. Computer, Vol. 26, No. 9, September 1993, pp. 8-22.

10. O'Brien, J. Hot Off the Wire. Computer Shopper 4, April 1994, page 603.

11. Schwartz, M.F. Internet Resource Discovery at the University of Colorado. Computer, Vol. 26, No. 9, September 1993, pp. 8-22.

Xplore As An Inexpensive Alternative

Carol A. Marinas
Asst. Professor of Mathematics
Barry University
11300 NE 2nd Avenue
Miami Shores FL 33161
305-899-3617
marinas@buvax.barry.edu

Special Thanks To Daniel Bulli, Barry University
Mathematics and Computer Science Major

History of Barry University and our Needs

Barry University is a 4 year private liberal arts university that was founded in 1940. Today our students number 6,500 and represent diverse religious, cultural, and ethnic backgrounds. In 1990, the Mathematics and Computer Science Department moved into a computer technology building that was funded partially by the Federal Aviation Administration. At that time, there was an extensive computer laboratory and 4 computer classrooms but the faculty offices had no computers. Since the distance between the faculty offices and these facilities were not relatively close, the need for computers in our offices became a priority. We were lucky; we got the rejected personal computers -- remember the ones without a hard drive that ran from floppies? The classrooms were better equipped than the instructors' offices! Finally in the Fall, 1991 the chairman persuaded the administration to supply the department of Mathematics and Computer Science faculty with 486 machines, including floppies and a reasonable size hard drive.

Now our next obstacle: software. Where do we get quality software for a network at a reasonable cost? We started by getting review copies of standard software packages and also software that was included free with the adoption of a textbook. We have considered many factors in our evaluation: cost, user friendliness, powerfulness, platform, and system requirements. In Appendix A, a brief comparison is given for X (plore), MathCad 5.0, Maple Release 2, Mathematica 2.2, and Derive 2.58. Although Xplore is not as powerful as the rest and not available as of yet in a Windows environment, we are very pleased with the friendliness, system requirements and the cost. With that in mind, let me tell you more about Xplore.

Xplore

X(PLORE) was written by David Meredith, a professor at San Francisco State University. For the PC version, a requirement of DOS 2.0 or higher and a minimum of 512K RAM is needed. X(PLORE) is a descendant of Calculus Calculator (Prentice Hall, 1991). While most files from Calculus Calculator can be used in X(PLORE), some of the commands have changed , therefore, requiring a few minor adjustments. X (PLORE) also contains more operations than its predecessor.

X(PLORE) is a powerful calculating environment for creating, exploring, and testing mathematical ideas. All tools in X(PLORE) are easy to access: differentiation, integration, sequences and series, matrices, vectors, and complex numbers. X(PLORE) offers both step-by-step calculations and programming, which can be used for complex algorithms.

The programming environment used in X(PLORE) is simple. Since commands are similar to the operations they perform, remembering these commands becomes easier.

Example : graph {SYMBOL 222 \f "Symbol"} for graphing
dif {SYMBOL 222 \f "Symbol"} for differentiation

If you have a question, on-line help is readily available.

Enclosed in Appendix B are different ways to lead a discussion of systems of equations using X(PLORE). The methods are graphing, solving algebraically, Cramer's Rule, row reduction, and inverse matrices.

Macintosh Upgrade

In March of this year, Xplore improved on its DOS PC version with a MacIntosh version. This program requires a MacIntosh computer with at least 500 KB of memory under System 6 or 7. Because the MacIntosh version contains a window environment, it is more user friendly than its DOS PC version. Commands are easily accessible by use of the mouse, but if preferred, these commands can also be typed in on the keyboard as in the DOS version. Unlike the DOS version, input, output, and graphics screen can all be viewed at the same time. Resizing each window to users' taste is also easily done.

Summary

With the advent of the MacIntosh version of X(PLORE), we will continue to use X (PLORE) for our entry level classes. Since our school has more personal computers, we have spoken to the author and publisher about the next step of a Windows version. The present status is: they need input for the demand before they authorize the programming. I am sure that with a excellent program and support from educators this will become a reality soon !

References

Derive Version 2.5, Honolulu, HI, Soft Warehouse, 1993.

Maple 5 Release 2, Waterloo, ON, Canada, Waterloo Maple Software Inc., 1991.

MathCad 5.0, Cambridge, MA, Microsoft Inc., 1994.

Mathematica 2.2, Champaign, IL, Wolfram Research Inc., 1993

X(plore) Version 4.0, Englewood Cliffs, NJ, Prentice Hall Inc., 1994.

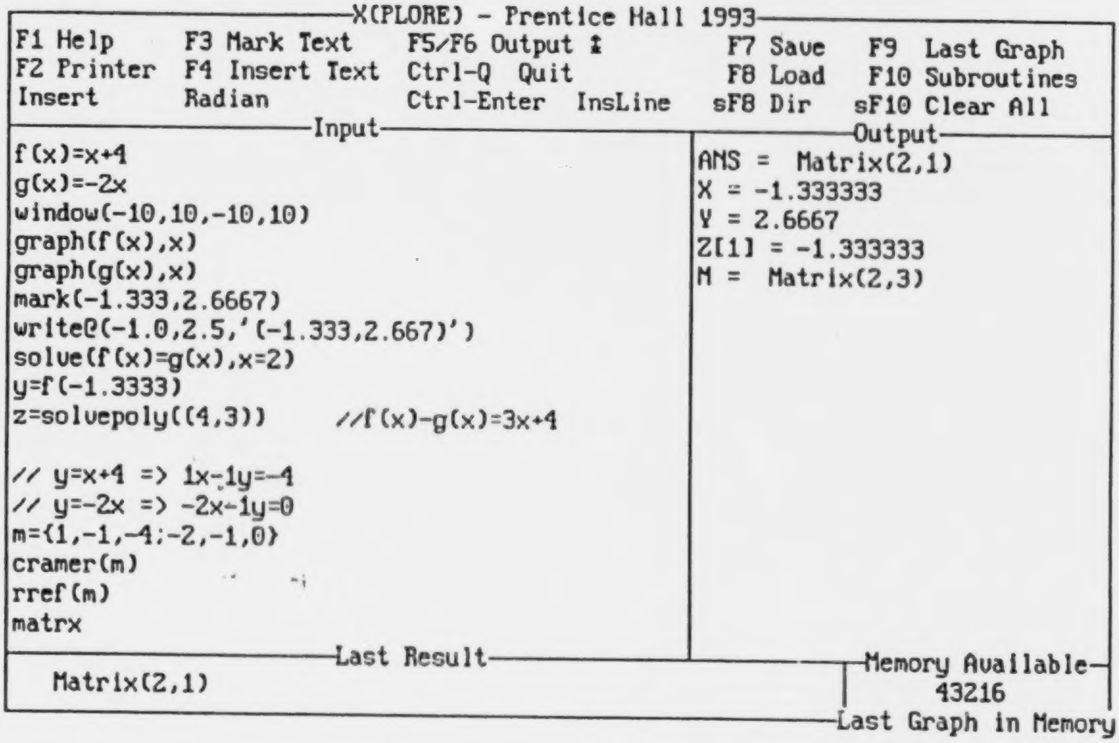
Kaplan, Gadi (November, 1993). "Math, visualization, and data acquisition," IEEE Spectrum.

Appendix A

Comparison of Xplore to Other Programs

	Xplore	MathCad 5.0	Maple V Release 2	Mathematica 2.2	Derive 2.58
Platform/ System	DOS 2.0 / 512 KB RAM/629K H.D MAC System 6 or 7 /500 K RAM	Windows 3.1 /8 MB RAM/ 14 MB H.D MAC version available	DOS 3.3/5 MB RAM/ 16 MB H.D MAC,UNIX, DEC VMS & AMIGA versions are available	DOS 3.0/6 MB RAM/13 MB H.D MAC System 7/ 6 MB RAM SUN version available	DOS 2.1/ 512 KB RAM/ 274 K H.D.
Friendliness	Xplore is very user friendly owing to the similarity between functions and their commands.	MathCad is complex and requires a vast amount of time and effort to use effectively.	Pull down menus are present	Pull down menus are present	Fairly easy to use with minimal practice.
Pros	Easy graphing, performs composition of functions, Has two demos for getting familiar with Xplore. On-line help is also available. Quickly Implemented for classroom use.	Symbolic Processor, real mathematical notation, text can be included along side graphics, on-line help available, and readily available electronic books. Tutorial available.	Has symbolic processor. Real mathematical notation. High level of mathematics. On-line help available. Factors and expands algebraic equations. Numerical & symbolic computation. Programming language.	Electronic books interface. Performs symbolic calculations. High level of mathematics. On-line help available. Numerical & symbolic computation. Excellent graphics with shading. Programming language.	Menu-driven symbolic math. Screens images are easy to print or store as tiff files. On- line help available. Factors and expands algebraic equations.
Cons	Not real mathematical notation (Fractional notation)	Extensive practice with MathCad tutorial to use features. Even with menus it is difficult to use. Excessive memory requirements.	Even with menus it is difficult to use. Excessive memory requirements. No tutorial present with software.	Even with menus it is difficult to use. Excessive memory requirements. No tutorial present with software.	Not real mathematical notation. No tutorial present with software.
Cost & Publisher Tel #	$30 .00 each Free with adoption Prentice Hall Inc.,Englewood Cliffs NJ George Lobell 201-592-2006	$59.50 each Mathsoft Inc., Cambridge, MA; 800-628-4223 ext. 732 Jim Gillespie 617-577-1017	$199.00 each Waterloo Maple Software Inc. Waterloo, ON, Canada 800-267-6583; 519-747-2373 800-628-4223 ext. 732 Jim Gillespie	$595.00 each Wolfram Research Inc., Champaign, IL 800-441-MATH	$375.00 each SoftWarehouse Inc., Honolulu, HI 808-734-5801

Appendix B


X(PLORE) - Prentice Hall 1993
F1 Help
F2 Printer
Insert
F3 Mark Text
F4 Insert Text
Radian
F5/F6 Output ↕
Ctrl-Q Quit
Ctrl-Enter InsLine
F7 Save
F8 Load
sF8 Dir
F9 Last Graph
F10 Subroutines
sF10 Clear All
Input
Output
Last Result
Memory Available
Last Graph in Memory

```
f(x)=x+4
g(x)=-2x
window(-10,10,-10,10)
graph(f(x),x)
graph(g(x),x)
mark(-1.333,2.6667)
write@(-1.0,2.5,'(-1.333,2.667)')
solve(f(x)=g(x),x=2)
y=f(-1.3333)
z=solvepoly((4,3))      //f(x)-g(x)=3x+4

// y=x+4 => 1x-1y=-4
// y=-2x => -2x-1y=0
m={1,-1,-4;-2,-1,0}
cramer(m)
rref(m)
matrx
```

```
ANS =  Matrix(2,1)
X = -1.333333
Y = 2.6667
Z[1] = -1.333333
M =  Matrix(2,3)
```

```
Matrix(2,1)
```

```
43216
```

Table 1

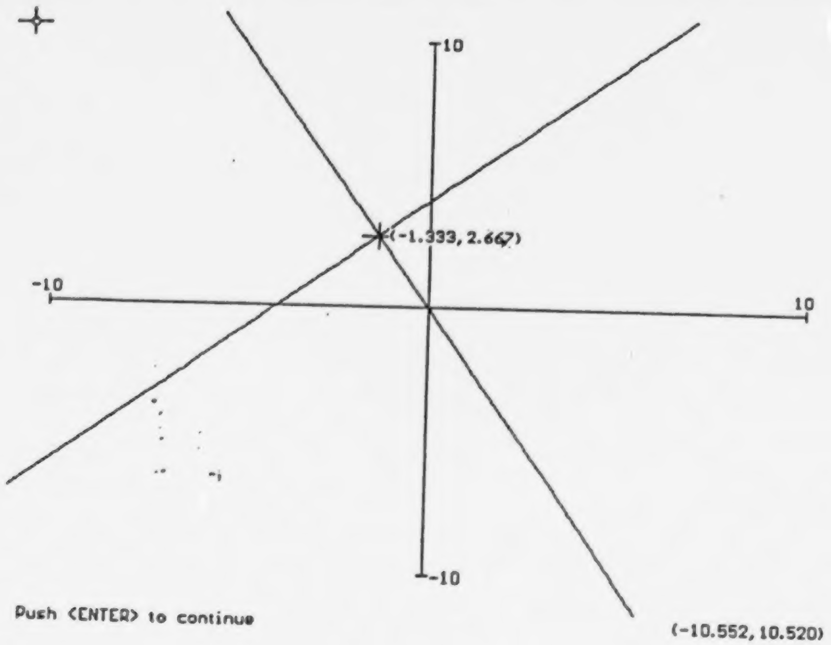
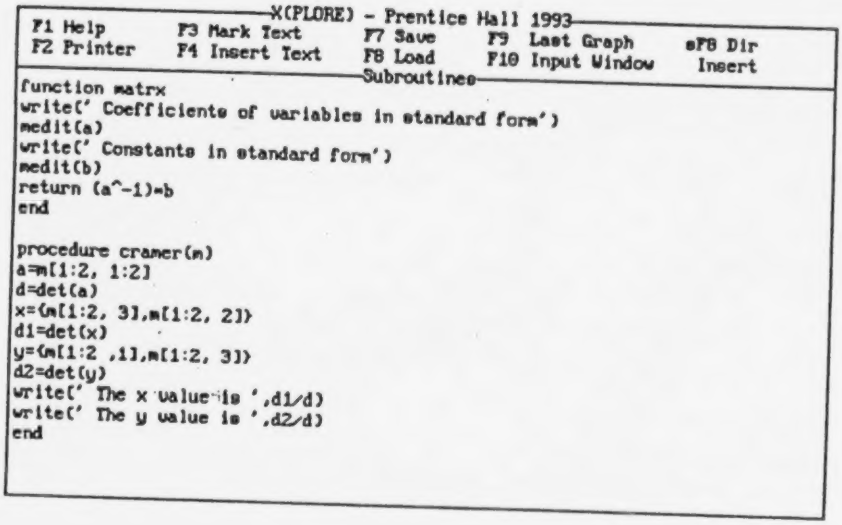


Table 2



```
X(PLORE) - Prentice Hall 1993
F1 Help        F3 Mark Text     F7 Save    F9  Last Graph      sF8 Dir
F2 Printer     F4 Insert Text   F8 Load    F10 Input Window     Insert
                                Subroutines
function matrx
write(' Coefficients of variables in standard form')
medit(a)
write(' Constants in standard form')
medit(b)
return (a^-1)*b
end

procedure cramer(m)
a=m[1:2, 1:2]
d=det(a)
x={m[1:2, 3],m[1:2, 2]}
d1=det(x)
y={m[1:2 ,1],m[1:2, 3]}
d2=det(y)
write(' The x value is ',d1/d)
write(' The y value is ',d2/d)
end
```

Table 3

Meeting the Theory Needs of Computer Science Majors

Small Computer Science Program
A. McCauley
Department of Computer Science
throp University
Hill, SC 29733
mccauley@lurch.winthrop.edu

This paper describes a course designed to provide students in a small undergraduate computer science program withthe theoretical foundations of computer science necessary for appreciation of the discipline and success in the curriculum.Due to the size and scope of our program, we considered a traditional, proof–oriented Theory of Computation course to beinappropriate for our students. Instead, we created a course which encompasses topics from both advanced discretemathematics and elementary computer theory, and emphasizes the practical applications of these concepts in the art andscience of computing. This course satisfies the recommendations of the Computing Sciences Accreditation Board and*Computing Curricula 1991: Report of the ACM/IEEE–CS Joint Curriculum Task Force.*

1 Introduction

The Bachelor of Science in Computer Science (BSCS) degree program at Winthrop University is accredited by theomputing Sciences Accreditation Commission (CSAC) of the Computing Sciences Accreditation Board (CSAB). Thecriteria used for evaluation of baccalaureate programs in the computing sciences are intended as guidelines for developingdegree programs that provide "an adequate foundation in science, mathematics, the humanities and social sciences, andcomputer science fundamentals, and to assure appropriate preparation in advanced computer science" ([1], page 36).valuation criteria relate to faculty requirements, curriculum, laboratory and computing resources, students, and institutionalsupport. While CSAB does not dictate curricula, they do specify characteristics that curricula should possess. The computerscience core material is divided into six areas in the CSAB literature: theoretical foundations of computer science, algorithms,data structures, software design, the concepts of programming languages, and computer elements and architectures. Thecurricular guidelines also discuss the roles of math, science, humanities, social sciences, arts, and other disciplines in thepreparation of computer science professionals [1].

In 1991, the ACM/IEEE–CS Joint Curriculum Task Force published a report called *Computing Curricula 1991.* Thiseport contains curricular recommendations for baccalaureate degree programs in computer science and computer engineering.Courses are not prescribed. Instead, these recommendations include a list of common requirements that constitute "a platformof knowledge that is considered essential for all students who concentrate in the discipline" ([2], page 21), as well asadvanced topics in computing, and math and science requirements. The common requirements are specified as a collectionof *knowledge units* considered to be fundamental to the discipline [2].

In our annual evaluation of the undergraduate CS curriculum at Winthrop University, we perceived a weakness inour program, despite the fact that the program had recently been re–accredited by CSAC. This perceived weakness was relatedto the coverage of some of the theoretical foundations of computing including the *Computing Curricula 1991* knowledge unitson: finite state automata and regular expressions (PL7), context–free grammars and push-down automata (PL8), and computability and undecidability (AL7). While these concepts were covered in our BSCS curriculum, they were scatteredacross several different courses including computer science and mathematics courses, we feared that students were not gettingthe "big picture" regarding the relationships among these topics. In many CS degree programs, these topics might be coveredin a traditional Theory of Computation course [6, 7]. However, we did not believe that a traditional, proof–oriented coursewas in the best interest of our students for a variety of reasons. One of those reasons is that most of our students enter thejob market immediately after graduation and only a small percentage of our graduates pursue graduate study in computerscience either immediately or eventually. We firmly believe in the fundamental importance of the theoretical foundations ofthe discipline in the preparation of computer scientists. We believe that the underlying theory is necessary in order for themto better understand and appreciate the discipline and also for them to be able to continue to grow and mature in thediscipline. Yet, we also believe that these concepts can be most effectively and appropriately presented to our students ina manner that related the theory to the practice of computing.

In this paper, we discuss the design and implementation of the course entitled "Theoretical Foundations" whichencompasses topics from both advanced discrete mathematics and elementary computer theory, and emphasizes the practicalapplications of these concepts in the art and science of computing.

2 Design and Implementation Issues

2.1 Rationale

Prior to the establishment of the "Theoretical Foundations" course, the BSCS degree requirements included twomathematics courses (six hours) covering topics typically referred to as discrete mathematics. (Typically, these courses aretaught in either mathematics or computer science departments.) The relevance of these types of courses in the undergraduatecomputer science curriculum is documented in several publications [3, 4, 5, 10, 11, 12]. Their use as a vehicle for introducingstudents to the theoretical foundations of computing has also been noted [8]. The first of these mathematics courses, entitled"Discrete Mathematics", remains a required course in our curriculum and is usually taken concurrently with CS1, which isthe first computer science course for computer science majors. CS1 teaches students to use computers to solve problems usinga high-level programming language. Both the Discrete Math and CS1 courses are prerequisites for CS2, the second problemsolving course for majors. The list of topics covered in the Discrete Math class is similar to the list suggested by Ralston[11] and includes: number systems, set theory, logic, inductive proofs, permutations and combinations, finite probability andrandom numbers, relations and functions, matrices and matrix operations, systems of equations, and boolean algebra.

The second course in discrete mathematics, which was entitled "Discrete Structures", covered advanced topics indiscrete mathematics, specifically functions and relations, graph theory, algebraic structures, and finite state machines andTuring machines. The content matter on finite state machines satisfied the Curricula 1991 required knowledge unit on finitestate automata and regular expressions (PL7). Finite automata and regular expressions were also covered in the senior-levelProgramming Languages course. The coverage of Turing machines in the Discrete Structures course was very brief and didnot include any depth of coverage on computability (AL7). The knowledge unit on context-free grammars/languages andpush-down automata (PL8) was covered in the Programming Languages course. The discrete structures course served as aprerequisite for the Programming Languages course. The problem that we perceived in this distribution of knowledge unitsacross several classes was that students were not comprehending the relationships between them.

So we chose to design a new course which would include both the advanced discrete math topics, as suggested in*Computing Curricula 1991*, and the "theory" units described above. While this would be somewhat of a reorganization ofmaterial currently covered in the discrete structures course, some of the topics previously taught in that course wouldnecessarily have to be eliminated.

2.2 Prerequisites

Considering the background we thought necessary and the advanced courses we thought might depend upon thiscourse, we decided that the first discrete mathematics course was a necessary prerequisite. Actually, we believe that thediscrete mathematics course is fundamental to all computer science courses. As mentioned above, at Winthrop Universitythis course is usually taken concurrently with CS1. This situation seems to be typical of many CS degree programs. We alsowanted to insure some computer science maturity of entering students because we thought that the most effective way topresent the theory-based material would depend upon the students' awareness of the nature of algorithms and certain datastructures, such as stacks and trees, which are taught in the CS2 course. That led to the inclusion of CS2 as a prerequisitefor the new Theoretical Foundations course. This makes the new course available to students in either their sophomore orjunior years. Our hope is that most will enroll in it during the second semester of their sophomore year. This will help themin many of the advanced courses that they will take during the last two years of their degree program.

The upper level course that most directly depends on the Theoretical Foundations course is the ProgrammingLanguages concepts course. This is because some of the topics that were previously taught in this course have been moveddown into the theory-based course. This should enable students to cover more of the advanced topics in the ProgrammingLanguages course.

2.3 Implementation

The topics specified in the syllabus for the "Theoretical Foundations" course are:

? relations, functions, and other foundations
? graph theory
? formal languages and computation theory
  - finite state automata, regular languages, regular expressions, regular languages
  - push-down automata, context-free languages, context-free grammars, parsing
  - Turing machines, phrase-structure grammars, turing acceptable and decidable languages, and
  - computability, decidability

We believe that the content of this course, while not altogether different from the original discrete structures course, is morelogical for majors and strengthens the theory component of our curriculum by laying the theoretical foundations required ofmany of the upper-division computer science courses in the curriculum and essential for success in the discipline.

A traditional Theory of Computation course would very likely cover all of the topics listed as pertaining to formallanguages and computation theory. It is likely that in a degree program offering a traditional, proof-oriented theory ofcomputation course, the prerequisites for such a course might include both a course in discrete mathematics and a course inprogramming languages, thus allowing a good portion of the semester to devoted to the topics related to Turing machinesand computability. [6,7] Clearly, our course is very different from this type of course.

For the purposes of accreditation, the Theoretical Foundations course is counted as consisting of 1 credit hour ofmathematics and 2 credit hours of computer science.

2.3.1 Implementation Details

The course was taught for the first time during the Spring semester of 1994. Some of the details of its initialimplementation follow:

The unit on relations, functions, and other foundations utilized approximately 7–9 class hours of the course, whichwas more than we expected it would. Some of this material was a review, but many concepts introduced were new to thestudents, including the notion of uncountable sets, the diagonalization proof technique, and the concept of uncomputablefunctions.

The graph theory unit included elementary concepts related to directed and undirected graphs and their practicalapplications in computer science, such as in the modeling of computer networks, scheduling problems and the controltructures of computer programs. Graph–theoretic topics covered included spanning trees, Euler paths, Hamiltonian circuitsand the travelling salesperson problem, graph traversals, and graph–coloring problems. Some of these topics, specificallyHamiltonian circuits and the travelling salesperson problem were included as vehicles for discussing "hard" problems,problems for which efficient computer solutions may not possible. Approximately 12 class hours of the course were devotedto graph theory.

The unit on formal languages and computation theory followed, utilizing about half of a semester. Specifically, thelanguage theoretical concepts were introduced in terms of recognition problems, and the computation theory in terms ofcomputability of functions. This material was new to virtually all of the students enrolled in the course.

Throughout the course, we emphasized the practical applications of the theory presented. Although, we used proofsto make points and lead students to certain conclusions, the students were not required to develop any complete proofs onthere own. This might be perceived as a weakness by some, but we were more concerned with the students' needs to use thelearned concepts to solve practical problems.

.2 Assignments and Student Evaluation

The bulk of the work done by students was on regular homework assignments which were discussed in class. Twoprogramming assignments were included in the course. In many similar courses, no programming is required. We decidedto include this minimal programming requirement for a couple of reasons. Firstly, the discrete structures course, as it waspreviously taught, always required students to implement a program related to graph theory so that they would have somexperience modeling practical problems using graphs. Secondly, as computer science educators, we commonly useprogramming problems to support the teaching of course content. Throughout the specifications of the details of theknowledge units in the common requirements listed in *Computing Curricula 1991*, laboratory experiences to support theteaching of the content materials are suggested.

The first programming assignment involved the manipulation of undirected graphs. The assignment was to determineif an arbitrary graph contained an Euler circuit and, if so, return one. The second assignment involved the development ofa Turing machine interpreter. This project incorporated software engineering concepts, many of which the students wereunfamiliar with. The students were divided into teams of three; each team was to develop its own Turing machine simulator.Most of the students had never worked on a team project before. The process they were to follow was spelled out clearly,including the development of a requirements specification document and, later, a design document. The specifics of whatthese documents were to included was also new to them. The documentation required of the working system was alsospecified. While the demands of this project were no where near that of a typical software engineering development project,we think the students benefited from the software engineering approach to the project. It was an attempt to move softwareengineering concepts into the undergraduate curriculum before the senior year, which is when the formal software engineeringcourse is usually taken. We also thought that the development of a Turing machine would help them better understand allof the abstract machines being discussed. The assignment was made about two weeks before Turing machines were discussedin class, so the students had to depend on published articles and their textbook to learn how these abstract machines worked. We believed that the entire experience related to this second assignment would be beneficial to their development ascomputer scientists.

Most of the daily homework assignments completed by students were discussed in class and not graded. A few ofhe assignments, including the two programming assignments were submitted for grading. Additionally, quizzes werefrequently given. Three tests, one covering the unit on relations, functions, and foundations, another on graph theory, anda third on the material related to regular and context-free languages, were given. The final examination focussed on thelanguage and computation theory elements of the course.

2.3.3 Textbook Selection

Finding an appropriate textbook for this course was difficult. We were not completely satisfied with any of theadvanced discrete mathematics/structures or traditional theory of computation texts that we found. We selected a new textentitled *Discrete Structures, Logic, and Computability* [9]. The book covered all of the theory units we needed and made agood presentation of most of the topics we covered. Throughout the text, the author emphasizes the significance of thepresented material in computer science. The book also includes many exercises and provided solutions to some of them inthe appendices. One disadvantage of this text for our course is its volume. It was written for a two-semester course and, thus,includes lots of material that we could not cover in our course. Because of its vast coverage of topics, we had to carefullywade through much of the material, covering some parts of some chapters, excluding others, and jumping back and forthamong chapters considerably. This seemed to be somewhat confusing to the students. Another disadvantage was that itscoverage of graph theory was minimal, in comparison to all of the graph–theoretic concepts that were included in the course.In the future, we may decide to follow the example of the book and reduce this content component of the class, as describedin the next section of this paper. If not, we will supplement the book with additional notes and resources in this area.

3 Course Evaluation

We strongly believe that the Theoretical Foundations course strengthens the theory component of our undergraduatecurriculum.

The course has only been taught one time. Therefore, we know that its execution can be improved upon. Firstly,we are considering reducing the amount of time spent on graph theory. Actually, there has been some discussion of movingsome of this material into the course we teach on algorithms and data structures, which many students take during the samesemester that they take the theoretical foundations course. However, currently this material is not being covered elsewherein the curriculum and we believe it to be important. The rationale for decreasing the amount of graph theory covered, is thatwe would like to have more time to concentrate on the language and computation theory units. As stated in a previoussection, this material is new to most students and our experience shows that they struggle with it quite a bit. We think thatadditional time to cover the material would be beneficial. Also, the computation theory is not covered elsewhere in ourcurriculum and we strongly believe that it is fundamental for computer science majors to have an understanding of what isnow or ever will be computable. Also under consideration is the elimination or reduction of the programming requirementsof the course. We do believe that appropriate programming assignments can strengthen the course, but also that programmingassignments may not always be the best way to teach all computer science topics. Similar courses often require weekly,graded homework assignments, rather than programming assignments and quizzes. We think that the Turing machineassignment was very appropriate and beneficial to this course.

The true evaluation of the course will occur next semester when these students enroll in the Programming Languagesconcepts course. It is expected that the students entering that course will be better prepared than those that have taken it inthe past. This should enable that course to cover more of the advanced concepts that the students need, since they will alreadybe familiar with the language theory concepts that course depends upon.

4 Conclusion and Future Directions

We have designed a course that we believe strengthens our undergraduate computer science curriculum and betterprepares students to enter either the computer science profession or graduate study in computer science. This curriculumchange was initiated in response to the requirements identified by the Computing Sciences Accreditation Board, *ComputingCurriculum 1991*, and published literature emphasizing the necessity of a solid foundation in mathematics and computertheory throughout the undergraduate curriculum.

The new course has only been taught one time. A few minor changes in the execution of the course are expectedfor next year. The next step in the strengthening of our curriculum may involve further analysis of the discrete mathematicscourse that is a prerequisite for the course discussed herein and further restructuring of other computer science courses tobetter meet the needs of our majors.

References

1. Computing Sciences Accreditation Board, Inc. (1993), *1993 Annual Report for the Year Ending September 30, 1993.*

2. Tucker, A.B. (editor) (1991), *Computing Curricula 1991: Report of the ACM/IEEE–CS Joint Curriculum Task Force.*

3. Marion, W. (1989), "Discrete Mathematics for Computer Science Majors – Where Are We? How Do We Proceed?," in *The Papers of the 20th SIGCSE Technical Symposium on Computer Science Education (SIGCSE Bulletin, 21(1))*, pp. 273 – 277, February 23–25, 1989, Louisville, Kentucky.

4. Henderson, P.B. (1990), "Discrete Mathematics as a Precursor to Programming," in *The Papers of the 21st SIGCSETechnical Symposium on Computer Science Education (SIGCSE Bulletin, 22(1))*, pp. 17 – 21, February 22–23, 1990,Washington, DC.

5. Bradley, J. (1988), "The Role of Mathematics in the Computer Science Curriculum," in *The Papers of the 19th SIGCSETechnical Symposium on Computer Science Education (SIGCSE Bulletin, 20(1))*, pp. 273 – 277, February 25–26, 1988,Atlanta, Georgia.

6. Bagert, D.J Jr. (1988), "The Increasing Role of Computer Theory in Undergraduate Curricula," Abstract of Panel Session, in *The Papers of the 19th SIGCSE Technical Symposium on Computer Science Education (SIGCSE Bulletin, 20(1))*, p. 223,February 25–26, 1988, Atlanta, Georgia.

7. Bagert, D.J. Jr. (1989), "A Core Course in Computer Theory: Design and Implementation Issues," in *The Papers of the20th SIGCSE Technical Symposium on Computer Science Education (SIGCSE Bulletin, 21(1))*, pp. 161 – 164, February 23–25,1989, Louisville, Kentucky.

8. Bagert, D.J. Jr. (1988), "Using a Course in Discrete Structures as an Introduction to the Theoretical Foundations ofComputer Science," *Proceedings of the 26th ACM Southeast Conference*, Mobile, AL, pp. 395–398, April 21–22, 1988.

9. Hein, J.L. (1994), *Discrete Structures, Logic, and Computability*, Jones and Bartlett Publisher, Inc.

10. Ralston, A. and M. Shaw (1980), "Curriculum '78 – Is Computer Science Really that Unmathematical?," *Communicationsof the ACM*, 23(2), pp. 67 – 70.

11. Ralston, A. (1984), "The First Course in Computer Science Needs a Mathematics Corequisite," *Communications of theACM*, 27(10), pp. 1002 – 1005.

12. Ralston, A. (1980), "Computer Science, Mathematics, and the Undergraduate Curricula in Both," *The AmericanMathematical Monthly*, 88(7), pp. 472 – 485.

Equipping the Classroom Data Jack

Michael R. Meeker
Department of Computer Science
University of Wisconsin Oshkosh
414-424-2068
meeker@vaxa.cis.uwosh.edu

Introduction

The trend to use computer technology as an integral part of course work has spawned the need to bring computers into all types of classroom settings. To respond to this trend, many colleges and universities have installing wiring systems to allow accessibility to LANs anywhere on campus.

Once the wiring system is installed, a new set of problems arise that center around what hardware and software should be acquired to best meet the highly diverse classroom needs of faculty. This paper addresses these problems by itemizing the issues that need to be considered and by surveying the suitability of available technology to meet these needs.

Classroom Design

Master Classroom

One approach is to design new types of classrooms around instructional technology. This approach leads to the creation of a 'master classroom' usually with a variety of technologies available for instructor and student use. When technologies (e.g. computer stations) are being provided for student use, such an approach seems justified. Such classroom designs often attempt to integrate analog and digital technologies into a multimedia system.

This approach has its strengths and weaknesses. The major strengths of the master classroom are:

- All or almost all technologies are available so the instructor may select the most suitable for the application.
- The multimedia presentations are more appropriate for today's visually oriented students.
- Non-sequential formats are possible.
- Master classrooms often provide for instruction to alternate between traditional classroom lecture and small group discussion/problem solving modes.
- The student is placed in an active role compared to the passive role assumed in the traditional lecture environment.

Some of the disadvantages of the master classroom are:

- The cost to equip such a classroom is between $35,000 and $50,000.
- Most designs limit class sizes to approximately 24 students, thus limiting accessibility to all students. For a campus of 10,000 students about 60-70 master classrooms would be required (assuming each could be fully utilized). The initial equipment cost would be $5 million to $7.5 million. These estimates do not include continuing upgrade and maintenance costs.
- Underutilization of equipment. The equipment is committed to a classroom, thereby rendering it unavailable to others even if it is not being utilized by the current classroom inhabitants.
- Classrooms may not be conveniently located near departments or other resources utilized by the course, e.g. laboratories, resource centers, etc.
- The faculty training and courseware development necessary to fully utilize a master classroom is extensive. Many faculty are not sufficiently convinced of the benefits of classroom technology and/or are unwilling to expend the effort.
- It is not clear that all subject matter requires such extensive technology for effective classroom delivery; in some cases technology may hinder rather than facilitate effective instruction.

Networked Classrooms

Another approach to utilizing instructional technology is to bring technology into existing classroom environments as needed. Equipment is strategically deployed to instruction environments only upon demand. Only a minimum of technology is permanently assigned to any particular location. In fact, the permanent assignment might be nothing more than a network terminus or what is often referred to as a 'classroom data jack'.

Like the master classroom there are strengths and weakness in this approach. Some strengths are:

- Only the needed technology is deployed; this simplifies the configuration and allows high equipment utilization levels.
- Classrooms may be close to department, laboratories, or other instructional resources.
- Costs are minimum because only a small amount of technology is permanently assigned to a given location. All other equipment is shared, resulting in reduced equipment inventory capital and maintenance costs.
- Large as well as small classrooms may be networked, thereby extending the use of technology to greater audiences.

Some often mentioned weaknesses are:

- The technology is limited primarily to instructor use, i.e. there is limited opportunity for students to utilize the technology.

- Existing classrooms often are ill suited for the use of visual systems. For example, it might be very difficult to create lighting conditions that allow a bright, clearly visible screen projection while at the same time provide students sufficient ambient light to take notes, refer to other materials, etc.
- Set up time is a significant limitation. Because equipment must be configured each time it is used, the complexity of the delivery system is restricted to configurations that can be interconnected, tested and initialized within a matter of minutes.
- Because equipment is shared amongst many classrooms, it is possible that, at peak demand times, some classroom needs will be unfulfilled.

Pedagogy

Lecture Mode

The commonest form of classroom activity in American higher education is the lecture mode. Whether one laments or extols the lecture, the fact is that there is little indication that lectures are in imminent danger of being replaced with other pedagogical methodologies. The programmed instruction movement of the early 1970's demonstrated how deeply rooted lecturing is in our educational system. Only a small amount of the vast amount of courseware developed for programmed instruction continued to be used by the end of the decade. If technology is to play a role in changing instruction it will probably have to do so by changing the lecture, not by replacing it. How then can technology be used to improve the lecture? The answer is at least two fold.

First, it is well known that there has been generational shifts in how students acquire information. Where earlier generations utilized printed media and then radio broadcasts, the contemporary college student uses active, integrated audio/visual media, i.e., television, cinematography, animation, etc. The key characteristics of these media forms are that they involve integrating two sensory modalities and that they depict motion or at least frequent change. When these characteristics are compared with the conventional lecture, it is relatively easy to identify several areas where technology could make lecture presentations more informational, animated, and interesting. Some of these are:

- Computer generated visual presentations can be made to be more accurate and attractive than blackboard or overhead illustrations prepared by hand.
- Change can be represented by presenting a series of states, each adapted from its predecessor. Preparation of such series would be far too time consuming to prepare without graphic presentation technologies.
- Videos and films can be inserted into lectures at intervals to better depict natural events and processes as well as to raise attention levels.
- Machine readable lectureware can be made available to students for review and reinforcement. The post-lecture availability of such materials lightens the student's note taking task, and hopefully allows greater attention to be focused on the presentation.

A second way technology can be used to enhance lectures is by providing access to multiple sources of information. In the traditional lecture, information is primarily disseminated orally. When other sources are used they are usually employed as substitutes, rather as augmentations, to the lecture, e.g. showing a film. This approach is not consistent with the way most students acquire information outside the classroom. For example, if information about a current event was desired, it is likely that the student would have multiple sources of information available. She/he might search through dozens of TV channels for coverage, telephone a publicly accessible information source, connect to a network (such as Internet, Prodigy, or CompuServe), or scan a newspaper. If classrooms are connected to networks, it becomes possible to access multiple sources of information during a single lecture. Some of the advantages of this approach are:

- Students are shown how to gather information from multiple sources in order to construct a more complete understanding of a topic than would be possible by consulting only a single source.
- The selection of information sources and/or options can be made by the student audience (with instructor guidance), thereby drawing them in as active participants in the lecture.

Perhaps the most significant change that technology brings to the lecture is to allow the focus to be placed on how to learn and acquire information rather than on factual monologues. This focus is well stated in the AASCU's Core of Academe:

"Professors may teach, but students learn. Further, students are quite capable of learning on their own. Given adequate resources from which to learn, the human organism is quite capable of self-instruction. The ability to learn independently, after all, is a valued quality of the professorate. Many educators think that the most significant single outcome of a university education is that students become lifelong learners when they leave the tutelage of the institution. The task of the professor, then, is to arrange the contingencies of teaching for learning to be most efficient and effective."

Demonstration

As instructors assign more outside activities involving computer methodologies, a growing need has emerged to teach students how to start using such systems. Simply telling students to go to a computer laboratory armed with a set of written instructions has proven to be inadequate, particularly for the student who has had little or no orientation to computers. Even when students have had previous computer experience, problems arise. For example, most of the secondary schools in Wisconsin use Macintosh systems because they have taken advantage of Apple Computer educational discount and grant programs. When these students encounter IBM PC systems without graphics interfaces, they are completely disoriented and require as much or more assistance than students with no computer background.

Bringing a computer system into the classroom, attaching it to a LAN and demonstrating system and application software protocols is an efficient and effective method for orienting students to campus computing facilities. Most students, having seen a classroom demonstration of what is supposed to happen in the laboratory, have sufficient orientation to perform the required laboratory tasks. However, several precautions should be observed:

- The classroom demonstration equipment and display should match, as closely as possible, what the student will see in the laboratory. Therefore, if high resolution, color displays exist in the laboratories, projection systems with similar capabilities should be utilized in the classroom. The general test of adequacy is that the instructor should never have to say 'although the screen looks like ... in the classroom, it will appear as ... in the lab' or 'although I am pressing ... to perform this function, you should ... in the lab'.

- The demonstrator should be careful to correlate his/her keystroke/mouse manipulations with consequent screen events. It is not sufficient to show that a piece of software can produce a particular result, the students needs to know how to get the software to produce that result. Although this point may seem trivial, it is surprising how easy it is to omit an explanation of keyboard/mouse manipulations. An effective technique is to state a desired result and ask the class what keystrokes or mouse commands are required to make this happen.

Interactive Classroom

One the most exciting new technologies with potential to significantly modify pedagogy is the response key system. Although the idea is not new, the technological implementation is. The technology is simple in concept. Each student is provided with a keypad which is connected to a computer system and display accessible to the instructor. The instructor can pose questions and ask all students to respond simultaneously. Answers can be in the form of multiple choice selection, true/false indication, or numerical values. Once students have responded the instructor can immediately see a profile of the responses and respond accordingly. If a significant number of responses indicate lack of understanding, optional explanation or instruction may be evoked. Another option is to allow students to direct the course of the classroom activities by polling them periodically. The interactive classroom differs from the usual in several significant respects:

- The students are transformed from passive to active classroom participants.
- All students participate, not merely the most verbal or knowledgeable.
- The instructor receives continuous feedback on student progress and understanding.
- Students receive immediate feedback on their responses as well as explanations of common misconceptions and misunderstandings.
- Nonsequential instructional formats become possible.

Computer Hardware

CPU/Memory

The software utilized by faculty for classroom teaching and demonstration is highly diverse and not restricted to any particular platform. Providing hardware that can adapt to everyone's needs, therefore requires the availability of powerful processors that can respond well to numerical, textual, file intensive, and graphic applications. It can be argued that the processor speeds and memory required for classroom computers should be at the high end of offerings because valuable class time is wasted if slow or small processors are utilized.

Instructors often wish to switch quickly between two or more operations. The systems should therefore be cable of supporting at least 3 concurrent tasks. Some examples of useful tasks configurations for classroom use might be:

- presentation software, word processor, library card catalog or journal index
- presentation software, data editor, statistical package
- cd rom data bank, spread sheet, internet connection

At the current time multiplatform systems are just being introduced. It may become possible that all needs can be met with a single system. However, at the present time it is probably necessary to provide both IBM PC and Macintosh processors to meet varying needs.

Secondary Storage

The amount and type of secondary storage that a classroom system should have is often a highly debated issue. These discussions revolve around several issues and concerns:

- Should every machine be capable of reading every manner and type of removable secondary storage that is available? If not, which media forms should be supported?
- Aside from operating systems and utilities, should sufficient secondary storage be provided to allow commonly used applications packages (e.g. word processor, spread sheet, data base) to reside on each system?
- Should courseware be pre-loaded onto systems that are delivered to classrooms, or should faculty have to load it from media they bring to the classroom or from a file-server on the local network?
- If systems are to be preloaded with software, how does the media resource center assure the integrity of the system and protect against deletion of modification of critical information?
- How do instructors transfer their courseware from a developmental site?

As can be seen from the nature of the above questions, the secondary storage questions lead to the need for policy development. Such policy should focus on assuring access to software/courseware and allaying faculty fears that lesson plans will be disrupted due to unavailability or corruption of resources.

As faculty develop multimedia presentations, the secondary storage requirements grow rapidly. If a number of faculty are sharing a single computer resource, it is highly possible that their combined needs will be able to be accommodated economically by dedicating secondary storage for each use. Further, tests at UW Oshkosh have indicated that transferring large files from a network file server to a local disk is so time consuming that it cannot be completed during the relatively small intervals available between classes. The conclusion is that removable secondary storage appears to be the only viable alternative that can comprehensively and economically address this problems. Fortunately there are several products on the market that would appear to meet the requirements. Removable hard drives are available from 250 MBytes through 540 MBytes; their cost starts at about $350. It is therefore possible to equip each classroom system with a docking module ($80), into which a hard drive may be inserted. If each developmental site (department or faculty office/lab) is similarly equipped with a docking module, faculty can transport their software/courseware about campus with ease. The main advantages of the removable hard drive approach are:

- Removable drives that are driven by an IDE controller achieve about 90-95% of the access speeds of non-removable drives.
- Operating systems, application software and courseware can be loaded onto a removable hard drive that is under the direct control of the faculty member. The question of maintaining system software integrity is therefore placed on the faculty member not the media resource center.
- Courseware development sites may be created inexpensively as $350 - $500 add-ons to existing systems.
- By adopting a campus standard for removable hard drives, department and other operating units can be assured that they can deliver, share, and develop instructional materials anywhere on campus.

Another form of secondary storage that should be strongly considered for inclusion on classroom systems is CD ROM. Much instructional software/courseware is being distributed in this form. In addition, CD ROM queries represent a way to introduce non-linear format into the classroom. Many computer systems configured for instructional purposes include CD ROM drives as standard or optional equipment.

Network Connection(s)

Today, most campuses utilize LAN's to support student laboratories. Therefore, any class that requires students to make use of these laboratories, has at least an occasional need to demonstrate network protocols and/or application usage. As sated above, the classroom demonstration equipment and display should match, as closely as possible, with what the student will see in the laboratory. This principle extends to the network protocol as seen by the user, i.e. logging on, logging off, menu options, etc.

The ideal network connection from classroom computer to LAN server is over a high speed dedicated data path. The amount of interaction between client and server precludes effective use of switched connections in most cases. Since wide area network capabilities are emerging rapidly, sufficient planning should be given to extending network connections beyond campus LANs.

Video Equipment

Evaluating and selecting appropriate classroom video equipment is one of the most challenging aspects of implementing networked classrooms. Often classrooms are not well designed for computer generated displays. The ambient light level necessary for students and instructors to take and read notes often diminishes the brightness of a screen display. Because of their highly concentrated light output, video projectors are least affected by lighting conditions. However, the best projectors are not portable, and those that are portable tend to have low resolution.

Video monitors placed throughout a classroom have also been used as video displays. However, the projected image is small and often unreadable unless viewed a short distance.

The most common form of projection unit today is the LCD panel. This unit, when placed atop an overhead projector, creates a relatively large, high resolution image. Compared with other display methods it has the greatest promise of producing acceptable displays. However, LCD panels have drawbacks that need to be considered:

- Projection of highly saturated displays result in only a small amount of light being transmitted through the panel. Even a small amount of ambient light can completely wash out such displays.
- High intensity overhead projectors of 4,500 to 6,000 lumens can partially compensate for the low light transmission through LCD panels. However, the light sources on these overhead projectors can shift the hues to create unnatural tones. This is particularly a problem when videos of natural phenomena are being displayed. In these cases, an LCD panel with tint control is mandatory. Not all LCD panels have operator controlled tint levels.
- If full motion video is to be displayed, the more expensive active panel displays are required.

Summary

The rapidly emerging demands for computer technology in the classroom are posing questions on how to effectively and efficiently deliver such technology anytime and anywhere on campus. A campus wiring system terminating in classrooms can serve as a basis for connecting to on- and off-campus resources. The joint requirements of portability and high reliability necessitate coordinated policy development and careful equipment selection. The technological offerings currently available reasonably address the needs and concerns of faculty.

An Expert System Prototype: Copyright Law in a Multimedia Environment

William J. Moressi
Academic Computing Center
Renee A. McCauley
Department of Computer Science
Winthrop University
Rock Hill SC 29733
803-323-2281
moressiw@winthrop.edu
mccauley@lurch.winthrop.edu

Abstract

Expert systems are software systems that emulate the problem-solving capabilities of human experts. Expert systems, commonly known as knowledge-based systems, are used in both the scientific and business communities.

This paper describes an expert system that encapsulates expert knowledge on copyright laws. Such laws now cover intellectual property in the form of printed materials and electronic data stored on magnetic or laser discs, magnetic tapes, and microfilm. Information on copyright laws and the protection of intellectual properties in various formats are essential as we begin to incorporate multimedia technology into our classrooms and offices.

Introduction

Our goals in developing this expert system prototype on copyright law were two-fold. Firstly, we wanted to create an electronic version of the information contained in the Winthrop University Policy on Copyrighted Material, so that we could make it easily accessible to the entire university community over the university computer network. Secondly, we wanted to create a system that would be useful as an instructional resource for teaching topics related to either copyright law or expert systems technology.

Expert systems technology is a successful area of study within the discipline of Artificial Intelligence (AI). Expert systems emulate the problem-solving capabilities of human experts. They are beneficial in that they make expert information available even when human experts cannot be present. Also, they can incorporate the knowledge of several experts into a single system and can usually explain the reasoning used in generating their conclusions. It is this property of expert systems that lends itself to our prototype project of explaining copyright law in a multimedia environment.

The area of multimedia involves many types of copyrighted works, the legal implications of which are overwhelming. Most information technology (IT) users have neither the knowledge of or expertise in copyright law to insure that they do not infringe on an author's property rights. This paper discusses copyright law, as it applies to the field of multimedia technology, and the development of an expert system that makes this expertise available "on-line".

Expert Systems

Expert systems (ES) are software systems designed to solve problems that are usually solved by human experts. Expert systems depend on a large body of stored knowledge, called a knowledge base, and a mechanism for reasoning with that knowledge, called the inference engine. Because of their dependence on large quantities of stored knowledge and knowledge processing techniques, expert systems are often referred to as knowledge-based systems. The knowledge base, the inference engine, and the user interface constitute a complete expert system.

Expert systems are usually limited to a narrow domain of expertise. Typically, they can respond to queries in the problem domain, provide explanations for answers generated, and sometimes offer alternative solutions to problems. Also, they are usually able to deal with probabilistic or fuzzy knowledge, by incorporating factors of uncertainty.

One of the best known expert systems, and one that is mentioned in virtually all AI and ES textbooks, is MYCIN. MYCIN encapsulates domain knowledge related to infectious blood diseases. Given the results of a patient's lab tests, MYCIN can diagnose and prescribe treatment and, when requested, it can explain its reasoning in English. MYCIN's success is comparable to that of human experts in the field of infectious blood diseases and was shown to outperform both medical students and general practitioners. Despite its remarkable success, MYCIN has never been put into practical use. Questions concerning the legal and ethical responsibilities for diagnoses and treatments precluded its practical use.

Since the emergence of MYCIN in the 1970's, many expert systems have been developed and used successfully in domains that are not "life critical" in nature. Some of the many fields in which knowledge-based systems technology has been applied are finance, manufacturing, airline scheduling, management, military science, geology, and software engineering. Knowledge-based systems have provided a variety of benefits including increased quality and speed in the performance of complex tasks, reductions in errors and costs, decreases in personnel, improvements in customer service, ability to combine knowledge from several experts, and the freeing of experts from making repetitive decisions. Expert systems technology is appealing because expertise is scarce and expensive. Additionally, as salaries for experts continue to rise, the cost/benefit ratio of developing expert systems continues to improve.

A person who develops an expert system is called a knowledge engineer. The first task of the knowledge engineer is to determine if the problem domain is suitable for the application of expert system technology. Expert systems will succeed in many, but not all, problem domains. Human judgements/conclusions are rarely based on knowledge alone, but also involve reasoning. Experts use intuition, insight, instinct, or rules of thumb that are not easily describable. ESs have not been successful in areas such as stock market forecasting, criminal investigations, and sports predictions. ES applications that are usually successful are those that are limited in scope and have domains that can be described in a straightforward manner.

Knowledge engineering is an iterative process that includes knowledge acquisition, system development, and system validation. Knowledge acquisition is the task of capturing domain-specific knowledge from an expert or group of experts. The task is difficult since experts are often unaware of all of the factors that they use when making expert judgements. Additionally, different experts may use different approaches to solving the same problem. The knowledge engineer must generate appropriate questions in order to elicit the information needed.

Before an expert system can be developed, the knowledge engineer must decide how the information will be represented and which programming language or system to use for the implementation. Most expert systems are developed using languages that can be described as rule-based. A knowledge base consists of facts in the domain and rules that specify how the facts may be used. Early expert systems were developed from scratch, many in LISP. Now, however, many expert systems are developed using software tools called expert system shells. An expert system shell provides the control mechanism for an expert system, i.e., the inference engine. The inference engine provides the "intelligence" of the system in that it determines how the facts and rules of the knowledge base will be used to draw conclusions or make decisions. The job of the knowledge engineer then becomes one of providing the domain specific knowledge. Use of these types of tools leads to quicker system development. Since such shells typically use a high-level, English-like language for rule construction, knowledge engineers need not be experienced computer language programmers. Additionally, expert system shells typically provide mechanisms for explanation, knowledge acquisition, and interaction with other programming environments.

The application described in this paper was developed using an expert system shell called VP-EXPERT. This package provides the inference engine, an editor to be used to enter the knowledge, and an English-like rule-based language to capture the knowledge. Additionally, this tool is able to give explanations for the conclusions it draws, incorporate uncertainty in terms of factors of confidence, interact with database files created by a variety database management systems, and interact with worksheet files created by a variety of spreadsheet programs.

Multimedia

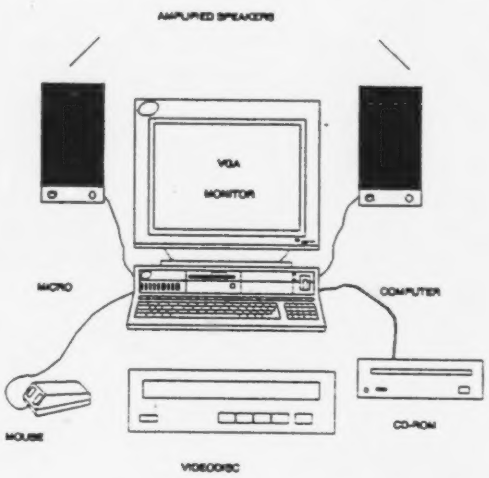
At present, there are as many definitions of a multimedia system as there are vendors selling them. Multimedia systems exist on multiple platforms. Essentially, a multimedia system allows one to communicate media (still, motion, audio, graphic and text images) in a computer based environment. Our concern in this paper is the media itself and the related copyright laws affecting it.

A typical multimedia system is represented in the following figure. How the elements of a multimedia system are packaged is determined by the specific end-user application. The elements must be configured in "technical harmony" so they work synergistically for the desired effect.

A basic system consists of a PC with CD ROM, an audio card and stereo speakers. This system is used mainly on an individual basis, by faculty or students who wish to incorporate graphics from the CD-ROM into their text files for handouts, reports, and other documents. It also suffices for small classroom presentations using an LCD viewer.

A mid-range system is comparable to many of the multimedia systems currently in place. In addition to the basic unit, it has a more powerful cpu, more hard disk space, higher quality sound card and speakers and may also come equipped with a videodisc player. It is most appropriate for use of interactive (canned) demonstrations and can be used to create and display animations. This setup is optimal for help centers and media centers.

MULTIMEDIA SYSTEM



A high-end system has an even more powerful cpu and increased storage capabilities. It has a digital audio card and a full-motion video card allowing for data capture and compression. This system allows one to combine video stills, full-motion video, audio, text, graphics, and animation into an interactive presentation. Opportunities abound for the use of these segments. They can be formulated into self-study programs, incorporated into tutorials, or set up for elaborate lecture presentations.

The more extensive and complex the media, the more complicated becomes the legal interpretation of copyright infringement. With the use of multimedia the potential for copyright violation is much greater. Many cases must be resolved in court. The following section on copyright law provides some insight into the law.

Copyright Law

© Copyright is a form of protection the law provides to the authors of "original works of authorship" for their intellectual works , both published and unpublished. The constitutional provision respecting copyright states "The Congress shall have power . . . To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the Exclusive Right to their respective Writings and Discoveries" (United States Constitution, Article I, Section 8). Although the rights provided by the law to the owners of the copyright are not unlimited in scope, it is illegal to violate any of these rights.

The Copyright statute, 17 U.S.C. § 101 et seq. (effective date: 1978), balances the author's interest against the public interest in the dissemination of information in areas of universal concern, such as art, science, history and business. The grand design of this delicate balance is to foster the creation and dissemination of intellectual works for the general public.

Copyright protection exists in original works of authorship fixed in any tangible medium of expression from which they can be perceived, reproduced or communicated either directly or indirectly by the aid of a machine or device. Our multimedia technology now brings this into special focus. Works of authorship include literary works (books and printed material); computer software; musical works (including accompanying words); video productions (motion pictures, videotapes); sound recordings; and dramatic works (plays).

Note, however, that statutory copyright protection does not include works that have not been fixed in a tangible form of expression such as titles, names, short phrases and slogans; works consisting entirely of information that is common property; and ideas, procedures, methods, concepts, principles, discoveries, systems, devices and processes.

The Copyright Act defines the rights of a copyright holder and how they may be enforced against an infringer. Included within the Copyright Act is the "fair use" doctrine which allows, under certain conditions, the duplication of copyrighted material. While the Copyright Act lists general factors under the heading of "fair use," it provides little in the way of specific directions for what constitutes fair use. The areas of primary concern to us are covered in sections 106 and 107 of the United States Code. Section 106 states the exclusive rights of the owner and section 107, the limitations on those rights, otherwise referred to as "fair use." The following excerpts are taken from Circular 92 of the United States Government.

106. Exclusive rights in copyrighted works

> Subject to sections 107 through 118, the owner of copyright under this title has the exclusive rights to do and to authorize any of the following:
>
> 1. to reproduce the copyrighted work in copies or phonorecords;
> 2. to prepare derivative works based upon the copyrighted work;
> 3. to distribute copies or phonorecords of the copyrighted work to the pubic by sale or other transfer of ownership, or by rental, lease, or lending;
> 4. in the case of literary musical, dramatic, and choreographic works, pantomimes, and motion pictures and other audiovisual works, to perform the copyrighted work publicly; and
> 5. in the case of literary, musical, dramatic, and choreographic works, pantomimes, and pictorial, graphic, or sculptural works, including the individual images of a motion picture or other audiovisual work, to display the copyrighted work publicly.

107. Limitations on exclusive rights: Fair use

> Notwithstanding the provisions of section 106, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include –
>
> 1. the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
> 2. the nature of the copyrighted work;
> 3. the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
> 4. the effect of the use upon the potential market for or value of the copyrighted work.

Although multimedia covers all areas specified above in "works of authorship", we will focus on three areas: Computer Programs, Video Productions (Motion Pictures and Videotapes) and Sound Recordings. The appendices provide definitions of terms and a list of U.S. Government publications on copyright law.

Computer Programs

The growing use and importance of computer software products raises difficult questions about how property rights should be protected. The proliferation of microcomputers, multimedia systems and both local and wide area networks have made the issues even more complicated.

In general, copyright law protects software against copying and distribution, even in the absence of a license agreement, unless it has been placed in public domain. This protection also extends to the documentation which accompanies most software; normally, it is illegal to make a copy of the complete software manual or documentation. Under the law of exclusive rights, only the owner of a copyright has the right to reproduce or distribute software or its documentation.

The following information on computer software is policy developed at Winthrop University based on an interpretation of copyright laws. The policy applies equally to all faculty, staff and students of Winthrop University. The policy includes all application and operating system software used on mainframes, mini-computers or microcomputers or any other computer devices, such as multimedia systems, using programmed software products.

The policy applies to the following categories of computer software:

a. Purchased Copies. This is defined as those software packages or programs which, when acquired, constitute an exchange of ownership from the seller to the purchaser.

b. Lease-only Versions. These are computer software packages or programs which require that a license fee be paid to the owner by the user. Such fee is considered as a lease payment, and ownership of the computer software or program remains with the seller of the license. These can be single copy versions, site licensed versions or network versions.

c. Evaluation or Demonstration Software. This is defined as those computer software packages or programs which are provided by the owner to the user for evaluation. In such case the prospective user is to use the software free of charge for a specified length of time. Ownership during this period remains with the seller of the license.

d. Shareware. This is defined as those unsolicited computer software packages and programs which are passed to a prospective buyer through the mail or by another individual. This software is to be evaluated, without contract, by the prospective buyer and if retained by the prospective buyer, the seller relies on a "good faith" transfer of a license fee from the user and to the seller of the license.

e. Public Domain Software. This is defined as non-copyright software which is offered for unrestricted use as public domain.

The following quotes define the limitations on exclusive rights regarding computer programs, 17 U.S.C. § 117. It is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

a. that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or

b. that such new copy or adaptation is for archive purposes only and that all archive copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

Any exact copies prepared in accordance with the provisions of this section may be leased, sold or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

Purchased Copies.

Copying for purposes of distribution whether it be for resale or for sharing is not permitted. Copying for archive purposes is permitted. Copying which is necessary to the utilization of the computer is permitted.

Lease-only Software.

Single Copy. Using a common copy of software on more than one microcomputer is not permitted, unless the lease agreement specifically states that this is permitted. Copying for purposes of distribution whether it be for resale or for sharing is not permitted.

Copying. Copying to an internal hard disk for purposes of installation is permitted, as is copying for archive purposes.

Network version. Network users are not permitted to copy software used in a network environment. The network administrator may make back-up copies of the software as required by the data network's recovery policy.

Site License. Copying for archive purposes is permitted at the user level. Copying with the intent to resell or share is not permitted. One serialized copy is permitted for use on one microcomputer. Use of a common serially numbered version on more than one microcomputer is not permitted.

Evaluation or Demonstration Software.

Evaluation/demonstration software can be requested for ordering through Dacus Library. Dacus Library will control the evaluation/demonstration software copies, providing for controlled check-out by faculty and staff for evaluation. Copying evaluation or demonstration software without the written consent of the rightful owner is not permitted.

Shareware

Ethical behavior is expected of faculty, staff and students with regard to shareware. If the software is retained or used, the requested license fee should be mailed to the rightful owner.

Public Domain Software

Software which is declared public domain is available for use by the public and therefore is exempt from copyright law.

Video Productions: Motion Pictures, Videotapes

The following guidelines reflect a national committee's (19 organizations appointed by Congress) consensus as to the application of fair use to the recording, retention and use of television broadcast programs for educational purposes. They specify periods of retention and use of such off-air recordings in classrooms and similar places devoted to instruction and for homebound instruction. The purpose of establishing these guidelines is to provide standards for both owners and users of copyrighted television programs.

a. A broadcast program may be recorded off-air simultaneously with broadcast transmission (including simultaneous cable transmission) and retained for a period not to exceed the first 45 consecutive calendar days after date of recording. Upon conclusion of such retention period, all off-air recordings must be erased or destroyed immediately. "Broadcast programs" are television or radio programs transmitted by television or radio stations for reception by the general public without charge.

b. Off-air recordings may be used once by individual teachers in the course of relevant teaching activities, and repeated once only when instructional reinforcement is necessary, in classrooms and similar places devoted to instruction within a single building, cluster or campus, as well as in the homes of students receiving formalized home instruction, during the first 10 consecutive school days in the 45-day calendar day retention period. "School days" are school session days--not counting weekends, holidays, vacations, examination periods, or other scheduled interruptions--within the 45 calendar-day retention period.

c. Off-air recordings may be made only at the request of and use by individual teachers and may not be regularly recorded in anticipation of requests. No broadcast programs may be recorded off-air more than once at the request of the same teacher, regardless of the number of times the program may be broadcast.

d. A limited number of copies may be reproduced from each off-air recording to meet the legitimate needs of teachers under these guidelines. Each such additional copy shall be subject to all provisions governing the original recording.

e. After the first 10 consecutive school days, off-air recordings may be used up to the end of the 45 calendar day retention period only for teacher evaluation purposes, i.e. to determine whether or not to include the broadcast program in the teaching curriculum, and may not be used in the recording institution for student exhibition or any other non-evaluation purpose without authorization.

f. Off-air recordings need not be used in their entirety, but the recorded programs may not be altered from their original content. Off-air recordings may not be physically or electronically combined or merged to constitute teaching anthologies or compilations.

g. All copies of off-air recordings must include the copyright notice on the broadcast program as recorded.

Sound Recordings

A sound recording is a work which may be fixed on a physical medium such as a phonorecord. The phonorecord may be a tape, cassette tape, cartridge or disk. In this section, reference to a phonorecord will be relative to that device upon which a sound recording is fixed.

The owner of a copyright of a phonorecord has the exclusive rights to distribute copies of the phonorecords of the copyrighted work to the public by sale or other transfer of ownership, or by rental, lease or lending 17 U.S.C. § 106(3).

Notwithstanding the provisions of 106(3), the fair use of the copyrighted work, including such use by reproduction in copies or phonorecords for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship or research, is not an infringement of copyright. Factors to be considered in fair use are were given earlier (§ 107. Limitations on exclusive rights).

The owner, by purchase or transfer of ownership, of a particular phonorecord obtained under section 106(3) is entitled without the authority of the copyright holder to sell or otherwise dispose of that phonorecord. The owner, however, may not dispose of directly or indirectly the phonorecord sound recording for purposes of commercial advantage by rental, lease or lending. Nothing in the preceding sentence shall apply to the rental, lease, or lending of a phonorecord for nonprofit purposes by a nonprofit library or nonprofit educational institution, 17 U.S.C. § 109(b)(1).

The Expert System Prototype: Copyright Law Advisor

Our expert system prototype, entitled "Copyright Law Advisor", encapsulates knowledge in the area of copyright law and intellectual property rights. The expert knowledge was derived from a document published at Winthrop University concerning the legal and fair uses of intellectual property across the campus. Since the expert knowledge had already been gathered, the job of the knowledge engineer became one of structuring the knowledge using the knowledge representation scheme supported by the expert system shell (VP-EXPERT).

The Copyright Law Advisor is menu-based. When the system is running, it presents the user with general information as to how to use the system. It then provides a menu of options from which the user can select an action. Specifically, the user can choose to read information on the fair use of copyrighted materials or the rights of owners of copyrights, to consult with the Copyright Law Advisor to learn how copyrighted materials can be legally used, or to exit from the system. A consultation is an interaction of a user with an expert system. A consultation begins when the user requests information from the system When the user chooses to consult with the Copyright Law Advisor, the system will prompt the him/her for the information it needs. This information will regard the type of intellectual work involved and the purposes for which the user may want to use or copy it.

The current version of the prototype incorporates knowledge on copyright law associated with computer software and data and video and sound recordings.

Conclusion and Future Directions

Our goals in developing this expert system were to create an electronic version of the information contained in the Winthrop University Policy on Copyrighted Material and to create a system that would be useful as an instructional resource. This prototype now includes copyright information regarding computer software, video productions, and sound recordings. We plan to enhance the system to include copyright information pertaining to other types of protected works. The Copyright Law

Advisor will be made available to campus personnel as an on-line reference source. For the study of expert systems and copyright law, appropriate instructional areas might include Computer Science or Management Information Systems.

References

El-Najdawi, M.K. and A. C. Stylianou. "Expert Support Systems: Integrating AI Technologies," Communications of the ACM, 36 (12), 1993, pp 55-65.

Firebaugh, M.W. Artificial Intelligence: A Knowledge-Based Approach, PWS-Kent Publishing Company, Boston, 1988.

Ginsberg, M. Essentials of Artificial Intelligence, Morgan Kaufmann Publishers, San Mateo, California, 1993.

Hayes-Roth, F. and N. Jacobstein. "The State of Knowledge-Based Systems," Communications of the ACM, 37 (3), 1994, pp. 27-39.

Hutching, Mary. "Model Policy Concerning College and University Photocopying for Classroom, Research, and Library Reserve Use," College and Research Library News, April 1982, pp. 127-131.

Moressi, Wm. J., B.G. Coxton, W.E. Hobgood, C.C. Hodgdon, and L.R. Mitlin. Winthrop University Policy on Copyrighted Material , Computer Utilization Committee, Winthrop University, Rock Hill, SC, 1991. (A copy of the document can be obtained from William Moressi.)

Munakata, T. "Commercial and Industrial AI," Communications of the ACM, 37 (3), 1994, pp. 23-25.

Schalkoff, R.J. Artificial Intelligence: An Engineering Approach, McGraw-Hill, Inc., New York, 1990.

APPENDIX A: DEFINITIONS

(Material extracted in whole or in part from Title 17 of the United States Code)

Audiovisual works are works that consist of a series of related images which are intrinsically intended to be shown by the use of machines or devices such as projectors, viewers, or electronic equipment, together with accompanying sounds, if and regardless of the nature of the material objects, such as films or tapes, in which the works are embodied.

Copies are material objects, other than phonorecords, in which a work is fixed by any method now known or later developed and from which the work can be perceived, reproduced or otherwise communicated, either directly or with the aid of a machine or device. The term "copies" includes the material object, other than a phonorecord, in which the work is first fixed.

Copyright owner, with respect to any one of the exclusive rights comprised in a copyright, refers to the owner of that particular right.

A work is created when it is fixed in a copy or phonorecord for the first time; where a work is prepared over a period of time, the portion of it that has been fixed at any particular time constitutes the work as of that time, and where the work has been prepared in different versions, each version constitutes a separate work.

A derivative work is a work based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation or any other form in which a work may be recast, transformed or adapted. A work consisting of editorial revisions, annotations, elaborations or other modifications, which, as a whole, represent an original work of authorship, is a "derivative work."

A device, machine or process is one now known or later developed.

To display a work means to show a copy of it, either directly or by means of a film, slide, television image or any other device or process of in the case of a motion picture or other audiovisual work, to show individual images non-sequentially.

A work is fixed in a tangible medium of expression when its embodiment in a copy or phonorecord by or under the authority of the author is sufficiently permanent or stable to permit it to be perceived, reproduced or otherwise communicated for a period of more than transitory duration. A work consisting of sounds, images or both, that are being transmitted, is "fixed" for purposes of this title if a fixation of the work is being made simultaneously with its transmission.

The terms including and such as are illustrative and not limitative.

A joint work is a work prepared by two or more authors with the intention that their contributions be merged into inseparable or interdependent parts of a unitary whole.

Motion pictures are audiovisual works consisting of a series of related images which, when shown in succession, impart an impression of motion, together with accompanying sounds, if any.

Musical works are works which include any accompanying words and are fixed in some tangible medium of expression. Musical works include both original compositions and original arrangements.

Phonorecords are material objects in which sounds, other than those accompanying a motion picture or other audiovisual work, are fixed by any method now known or later developed, and from which the sounds can be perceived, reproduced or otherwise communicated, either directly or with the aid of a machine or device. The term phonorecords includes the material object in which the sounds are first fixed.

Pictorial, graphic and sculptural works include two-dimensional and three-dimensional works of fine, graphic and applied art, photographs, prints and art reproductions, maps, globes, charts, diagrams, models and technical drawings, including architectural plans. Such works shall include works of artistic craftsmanship insofar as their form but not their mechanical or utilitarian aspects are concerned; the design of a useful article, as defined in this section, shall be considered a pictorial, graphic or sculptural work only if and only to the extent that, such design incorporates pictorial, graphic or sculptural features that can be identified separately from, and are capable of existing independently of the utilitarian aspects of the article.

Publication is the distribution of copies or phonorecords of a work to the public by sale or other transfer of ownership, or by rental, lease or lending. The offering to distribute copies or phonorecords to a group of persons for purposes of further distribution, public performance or public display constitutes publication. A public performance or display of a work does not of itself constitute publication.

To perform or display a work publicly means:

1.to perform or display it at a place open to the public or at any place where a substantial number of persons outside of a normal circle of a family and its social acquaintances is gathered; or

2.to transmit or otherwise communicate a performance or display of the work to a place specified by clause or to the public by means of any device or process whether the members of the public capable of receiving the performance or display receive it in the same place or in separate places and at the same time or at different times.

Sound recordings are works that result from the fixation of a series of musical, spoken or other sounds, but not including the sounds accompanying a motion picture or other audiovisual work, regardless of the nature of the material objects, such as disks, tapes or other phonorecords, in which they are embodied.

A transfer of copyright ownership is an assignment, mortgage, exclusive license or any other conveyance, alienation, or hypothecation of a copyright or of any of the exclusive rights comprised in a copyright, whether or not it is limited in time or place of effect, but not including a nonexclusive license.

A computer program is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result.

A transmission program is a body of material that, as an aggregate, has been produced for the sole purpose of transmission to the public in sequence and as a unit.

To transmit a performance or display is to communicate it by any device or process whereby images or sounds are received beyond the place from which they are sent.

Video productions are works which include motion pictures and videotapes. "Motion pictures" are audiovisual works consisting of a series of related images which, when shown in succession, impart an impression of motion, together with accompanying sounds, if any. "Videotape" means the reproduction of the images and sounds of a program or programs broadcast by a television broadcast station licensed by the Federal Communications Commission, regardless of the nature of the material objects, such as tapes or films, in which the reproduction is embodied.
A work made for hire is:

1. a work prepared by an employee within the scope of his or her employment; or

2. a work specially ordered or commissioned for use as a contribution to a collective work, as a part of a motion picture or other audiovisual work, as a translation, as a supplementary work, as a compilation, as an instructional text, as a test, as answer material for a test, or as an atlas, if the parties expressly agree in a written instrument signed by them that the work shall be considered a work made for hire. For the purpose of the foregoing sentence, a "supplementary work" is a work prepared for publication as a secondary adjunct to a work by another author for the purpose of introducing, concluding, illustrating, explaining, revising, commenting upon, or assisting in the use of the other work, such as forewords, afterwords, pictorial illustrations, maps, charts, tables, editorial notes, musical arrangements, answer material for tests, bibliographies, appendixes and indexes. An "instructional text" is a literary, pictorial, or graphic work prepared for publication and with the purpose of use in systematic instructional activities.

Appendix B: COPYRIGHT BIBLIOGRAPHY

U.S. Copyright Law. Title 17 of the United States Code (revised to March 1, 1989). Circular 92. Washington, D.C.: Government Printing Office, 1989.
U.S. Copyright Office. Library of Congress. Circular 1, Copyright Basics. Washington, D.C.: Government Printing Office, 1988.
U.S. Copyright Office. Library of Congress. Circular 2, Publications on Copyright. Washington, D.C.: Government Printing Office, 1988.
U.S. Copyright Office. Library of Congress. Circular 7d, Mandatory Deposit of Copies or Phonorecords for the Library of Congress. Washington, D.C.: Government Printing Office, 1989.
U.S. Copyright Office. Library of Congress. Circular 21, Reproduction of Copyrighted works by Educators and Librarians. Washington, D.C.: Government Printing Office, 1988.

U.S. Copyright Office. Library of Congress. Circular 22, How to Investigate the Copyright Status of a Work. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular 40, Copyright Registration for Works of the Visual Arts. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular 40D, Copyright for Audiovisual Material. Washington, D.C.: Government Printing Office, 1974.

U.S. Copyright Office. Library of Congress. Circular 49, Copyright for Audiovisual Material. Washington, D.C.: Government Printing Office, 1977.

U.S. Copyright Office. Library of Congress. Circular 50, Copyright Registration for Musical Compositions. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular 56a, Copyright Registration of Musical Compositions and Sound Recordings. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular 60, Copyright for Books. Washington, D.C.: Government Printing Office, 1977.

U.S. Copyright Office. Library of Congress. Circular 61, Copyright Registration for Computer Programs. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular 67, Poems and Song Lyrics. Washington, D.C.: Government Printing Office, 1977.

U.S. Copyright Office. Library of Congress. Circular 96, sections: 202.19, 202.20, and 202.21. Washington, D.C.: Government Printing Office, 1989.

U.S. Copyright Office. Library of Congress. Circular R15, Extension of Copyright Terms. Washington, D.C.: Government Printing Office, 1983.

U.S. Copyright Office. Library of Congress. Circular R45, Copyright Registration for Motion Pictures Including Video Recordings. Washington, D.C.: Government Printing Office, 1980.

U.S. Copyright Office. Library of Congress. Circular R49, Copyright Registration for Video Games and other Machine Readable Audiovisual Works. Washington, D.C.: Government Printing Office, 1985.

U.S. Copyright Office. Library of Congress. Circular R99, Highlights of the Current Copyright Law. Washington, D.C.: Government Printing Office, 1985.

U.S. Copyright Office. Library of Congress. Circular R96, section 201.18. Washington, D.C.: Government Printing Office, 1985.

U.S. Copyright Office. Library of Congress. Circular R96, section 201.18. Washington, D.C.: Government Printing Office, 1984.

U.S. Copyright Office. Library of Congress. Circular R96, section 202.10. Washington, D.C.: Government Printing Office, 1985.

U.S. Copyright Office. Library of Congress. Circular R96, section 202.1. Washington, D.C.: Government Printing Office, 1984.

Personnel - The Key to Making It Work

Warren Murphy
The Library --- Wallace Hall
Troy State University
Troy AL 36082
205-670-3250
wmurphy@asntsu.asn.net

In the effort to stay current and take advantage of developing technology, institutions are faced with decisions of how to use the available technology, how much it will cost and who can implement what is available. The "who" can be the key to decisions.

Normally there are three alternatives to the personnel problem of taking advantage of technological advances. An institution can have consultants make recommendations about how to proceed and provide designs to be implemented by the consultant's organization or by an independent contractor. As an alternative the institution could enter into an agreement with a single vendor to make recommendations and implement a plan of action. Another option would be for the institution to acquire personnel with the necessary expertise to do the design and implementation of technological advances to meet the institution's unique requirements.

The concept of using a consultant is very popular. The thought of a knowledgeable, experienced individual surveying the existing conditions and reviewing the institution's planning statements (they do exist in detail, right?), along with interviewing interested parties appears to be a reasonable approach to getting something done right the first time. When the decision is made to use the services of a consultant someone must make a selection. Is the decision made by an individual or by committee? Either way, are the personnel involved qualified to make the selection? How will the selection be made? Will it be someone you know, someone recommended by colleagues on your campus or other campuses, or someone identified in a literature search? The same search fundamentals as those used for selection of new faculty or staff members will be necessary: recommendations accepted, contacts made, references checked - we are talking about time-consuming effort here. The time required and in-house staff involvement will not stop with the selection of a qualified candidate or firm. The consultant can only make decisions based on what he/she observes or from information gathered in interviews. Someone must be assigned to work with the consultant and supervise tours or schedule meetings with all departments or functional areas on campus. Who has the time? When the consultant's work is complete, someone must interpret the recommendations and manage or supervise the implementation or partial implementation of the recommendations within the ever-changing campus environment.' Will a person or an entire staff be required to work with vendors and installers and approve work completed? Maybe those duties will be assigned to existing personnel who are already overburdened. Either way we approach the issue, there will be a large amount of staff time required to prepare for and choose a consultant, work with the consultant during the evaluation process, and implement the consultant's recommendations. Can the time be better spent?

When the consultant's recommendations are implemented, can one vendor provide all of the components and/or personnel to complete the project or a well defined portion of the project? It is very common for more than one vendor to be involved. Will a single vendor assume the responsibility for full implementation and use subcontractors for components outside their domain? How will these considerations effect the bidding process? If a one vendor solution can not be identified, in-house personnel must search for qualified vendors to complete the entire project.

When the selection of vendors and the single vendor versus multiple vendor concept is considered, we must look seriously at the working relationships which will be required with the vendors. If these working relationships must exist, we might ask ourselves, "Why deal with a consultant?". Rather than use a consultant, an institution might choose to use the services of a vendor for the evaluation and recommendation process, as well as the implementation. In this case the vendor assumes the role of the consultant. There is a notable exception which must be addressed at this point. The vendor may suffer from tunnel vision with respect to a problem solution. The vendor's experience may be limited to a particular product line or the concept of "canned" business solutions. A vendor with a reputation for providing consulting services which cross company and multiple technology boundaries is a viable solution. One very positive aspect of this type solution is the turn-key system concept. From a service and support viewpoint, this usually yields a single point of contact for repairs, upgrades, and development of growth paths.

Regardless of which combination of the consultant and vendor solution is applied, some things remain constant. In-house personnel must provide the fundamentals. Who is the customer? What product is delivered? What are the priorities? Personnel time, talent, and treasure (TTT) will be required in the evaluation, implementation, and support processes when these questions are answered. After implementation of the project, how much time, talent, and treasury will be required for oversight of operations even though product support and growth planning must come from outside sources? How long will the people involved in the planning and implementation of your project be available? The time will come when you hear, "Leroy did that. He is no longer with us" or "Sue would know if she were here". Will in-house personnel be able to maintain continuity. What will be the cost of someone learning a system from the basics?

As we examine these possibilities for implementation and use of rapidly changing technology, a constant is the requirement of in-house personnel. When this requirement is recognized, should the commitment be made to acquire the personnel capable of the evaluation, recommendation, implementation, and continued oversight of a solution? This is a major step. Decision makers must realize in the beginning that one person can not do the job. Staff will be needed, but one person will be the key. This key person would provide training paths and mentoring for targeted talented personnel to provide long term continuity to the system support. Now we must ask, "what are the qualifications of such a person"? One person will be asked to communicate with the departments and functional areas within the institution. Their specific needs must be identified and evaluated. Products necessary to meet these specific needs must be chosen and vendors identified who can provide quality products. The installation of the products must be inspected and approved to be within specifications. This person, with the assistance of staff, will provide operational training, structure for the operation, and continued support of the systems. To accomplish this the key person must possess experience, knowledge, and awareness.

The individual must have experience in the technologies required to build efficient, multi-faceted systems. Computer hardware and software experience will be required, along with experience in data communications and telecommunications. Experience with both analog and digital solutions to problems will be a benefit. Knowledge of various hardware architectures and the implementation of different operating systems on those architectures will be required. Knowledge of systems engineering, data engineering, and software engineering will be required. Facilities management and the various aspects of security (hardware, software, personnel, facilities) must be understood. Knowledge of the environment in which the systems will operate will is mandatory, as well as knowledge of the engineering principles of design, process, and directing/controlling the process. A working knowledge of mathematics and statistics principles is essential for creating models and simulations. This working knowledge will also be used in the gathering and evaluation of the metrics necessary to provide efficient and effective systems.

With the necessary experience and knowledge required to accomplish the desired goals, this individual must be aware that all answers or solutions are not technical. A full awareness must exist for the need to bring other people into the decision-making process. An awareness of budgeting processes and cost accounting need to be combined with contracting and legal issues, business ethics and human resources. One person cannot possess all of the skills and knowledge required to meet the challenges. Colleagues must be identified and a strong team of diverse talent be assembled to guide any implementation effort. Strong communications skills, both oral and written, will be required by this key person for effective exchange of information with colleagues and the management involved.

Are people available who meet these requirements? The answer is, "Yes". In most cases, these people will have a strong engineering background, even though they may not be engineers. The existence of these people may be pursued by examining where the discipline related to information systems is today and how it got where it is. When one considers that the first commercial use of a computer occurred only forty (40) years ago, we see that this is a young discipline. In the early days of efforts related to information systems, most of the people associated with computers were engineers and scientists. Because of the nature of machines at that time, and the way software had to be used on the machines, these engineers and scientists had to continue to work with the machines even in the commercial environment. This required that the "computer people" understand the particular commercial application. A very close working relationship existed between the "computer people" and experts associated with the commercial application.

These computer people were knowledgeable about the machines and the software which was implemented on each machine. This was required because the solution to any given problem might require that changes be made in the hardware, as well as the software. As the technology progressed and hardware/software solutions moved away from batch applications, the concepts related to printing terminals and then display terminals became reality - i.e. data communications. As high level languages (FORTRAN, COBOL) were developed and gained wide acceptance, the ability to be productive "computer people" became farther removed from the machines and the operating systems which had been implemented. It was the introduction and acceptance of these high level languages that prompted the discipline to progress at the unbelievable pace which has been experienced. As that progress was being made, so were improvements of speed and performance in hardware and in the sophistication of operating systems. This was guided by the original engineers and scientists who first made use of computers. As work progressed, people working with and for these pioneers became very knowledgeable about all aspects of computers; detailed knowledge of hardware, of operating systems, and data communications. These people are knowledgeable about the electronic technology used in building computers and in the software, from machine language forth, needed to accomplish specific tasks on the machines. When problem solutions are sought the scope of possibilities is as large as current technology permits. These individuals are aware of the necessity of cooperation among disciplines to achieve problem solutions. They will be among the first to advocate team solutions and admit that no one can stand alone. Considering the age of the discipline, it can be seen that these people are in the market place today. If interviewed, these individuals may or may not know the latest buzz words, but given the opportunity, they will be able to explain the operations of computers and communications in great detail, probably more boring detail than most people want to hear.

As these key people and their assembled teams move to take advantage of technological advances, assistance may be required from consultants and vendors to achieve the desired goals. It can be seen that all of these players serve a vital function in this ever-changing environment. Which ever approach is taken to achieve the goals, a single point of contact and responsibility should be maintained to assure continuity for support over the long term. The driving force of how the process takes place should be driven by the fundamentals. Who is the customer(user)? What product or service is delivered? What are the priorities?

Computer Competencies: Setting a Quality Standard

TQM Team #3
Janet Polansky
University of Wisconsin-Stout
P.O. Box 790
Menomonie WI 54751-0790
715-323-1103
polansky@uwstout.edu

University of Wisconsin-Stout, a four-year professional-technical college in Wisconsin's comprehensive system, emphasizes, according to its mission, academic programs which address "the evolving economic, technical, environmental ... demands of a global society" which "integrate technology, business, and service concepts" and experiment with "new instructional methods." Although the university's mission is also to offer "Classrooms and laboratories equipped with modern learning technologies and up-to-date equipment," classroom teachers and support staff involved with student computer use in the classroom and in laboratories encountered widely differing levels of computer competency when academic tasks in general studies and introductory courses. Computer skills are not a universal requirement in the curriculum, and no consensus exists as to what level of general skills and abilities are necessary for success in academic coursework and in the entry-level job market. The institutions' yearly survey of recent graduates and their employers provided feedback on the effectiveness of its programs in a number of areas identified as necessary to professional success, but computer use was not among them.

In view of these needs and of education's long-term goals of providing a technologically literate workforce and society, the university's Quality Leadership Council, upon the suggestion of instructional faculty and staff, made computer competencies one of its quality management projects for 1993-94. A Provost-sponsored TQM team was to design a model of computer competence necessary for success in general university coursework and upon entry-level employment and made recommendations about the university's present capability and future needs.

The Team and Its Charge: The team, comprised of five teaching staff, of whom one was to be team leader, the director of the major campus computer lab, a supervisor of custodial staff who was designated as the group's facilitator, and two students, was to:

1) describe the ideal model of student computer competency using organizational goals, relevant discipline, and customer needs data gathered from students, program directors, and employers and, in view of this,

2) assess the present design and its problems, adjust for contingencies, and
3) establish an action plan, milestones, and ownership for a new standard (i.e. model) which would be communicated to the various stakeholders involved in the process.

The Process: The team, whose members represented a wide array of perspectives, was made up not primarily of computer experts, although several members had very high expertise or even taught computer skill, but of community members who worked with students' computer skills in a variety of ways. And although several members had a great deal of interest in quality management, only the team leader had much prior experience in quality teams. The team was trained in a two-day workshop conducted by Casey Collett, during which they established their objectives (design vs. action), set their rules, planned for data collection. Although several universities have computer skills in their curriculum, none ties this integration to levels of computer competency. A design for doing so would prove to be a benchmark.

The project was somewhat of an experiment in the sense that, although the campus had witnessed several quality action team in the past two years during which TQM had become a campus hot-button, a certain amount of skepticism existed about the extent to which TQM, viewed as primarily a management tool used to correct defects in a clearly identifiable and often mechanical process, could be applied to educational problems, especially complex ones. On the other hand the team, which quickly formed a bond, felt more confidence than an ordinary task force or research group might in its ability to make a direct impact, since so many of its members had direct responsibility for computer training or utilization, and that the key to its success was in its ability to communicate its results in many directions and many channels.

Literature Review: Current wisdom suggests that computer literacy is no longer an option in today's job market. The role of computers in the workplace has expanded in recent years and it has become evident that experience in computer use will be a critical factor in achieving success in many fields, and that students graduating from colleges and universities would need skills in microcomputer applications in their chosen career field (Smith and Furst-Bowe, 1992). Business and industry were forming working relationships with learning institutions to provide real-world computer training, and the number of students, employees and instructors requiring computer software training was growing at an enormous rate.

Since much has been written about the use of computers as teaching tools in higher education and as productivity tools in business and industry, educators seem to have abandoned the debate over whether computers should be introduced into the classroom and to have turned their focus on the questions of how computers can be used effectively in education (Lloyd and Gressard, 1984). Although computer skills among faculty members in higher education vary greatly, most instructors expected their students to have some degree of computer literacy when they enter the college classroom, (Hirschbuhl and Faseyitan, 1994).

Employers also appear to expect newly-hired graduates to possess computer skills. In 1991, the Secretary's Commission on Achieving Necessary Skills (SCANS) identified several competencies needed for young people to compete in the labor market. These competencies included selecting computers systems, operating computers, using computers to process information and solving problems with computer technology (The Secretary's Commission on Achieving Necessary Skills, 1991). According to the Commission, computational skills will be essential as virtually all employees will be required to maintain records, estimate results, use spreadsheets or apply statistical process controls as they negotiate, identify trends or suggest new courses of action.

However, although both college faculty members and employers expect students to possess computer skills, little research had been conducted to identify student skill levels in computer use as they enter or as they graduate from post-secondary institutions. Many factors could produce wide variations in the computer skill levels of college students. The computer skills of college students may vary depending on the courses they completed in high school and college, their academic major, the work experience they have had and their personal interest in computers and computing (Smith and Furst-Bowe, 1992).

Some universities have conducted research to measure students' computer literacy, which refers to student's knowledge about computers and computer operations. Other universities have studied how computer literacy affects student's ability to use computers in the classroom and in the workplace (Strickland, 1989). In a study at the University of Virginia, it was found that levels of computer experience varied greatly among college students. This study indicated that students with more computer experience were significantly more confident in their use of computers that were students with little experience (Lloyd and Gressard, 1984). A study conducted at Purdue University also tried to measure students' knowledge of and previous experience with computers. This study attempted to gain a systematic knowledge of student's computer background. The researcher found that although half the students surveyed had previous experience with computers, many of students did not use computers regularly and had no motivation for using computers in their classes. It was recommended that more complete knowledge about students' computer literacy could help educators eliminate conception about students' abilities and develop curricula that more closely meets the needs of students (Sullivan, 1989).

Since 1985, students at Bentley College have been completing a questionnaire regarding their computing experience. The major trend indicated by these annual surveys is that the number of students with a background in programming is decreasing; however, the number of students with some background in productivity software, such as word processing, spreadsheet, and database management, is increasing. The results of this study have been used to tailor the school's computer courses to students' backgrounds (Harrington, 1990).

Finally, a study at the University of Wisconsin-Stout measured the computer skills of students in the fall of 1992. The results of this study suggested that the vast majority of students knew how to use computers and knew how to use word processing software. However, approximately one-half of the students had used spreadsheet or database management software and approximately one-third reported experience with computer programming. The majority of respondents reported that they had obtained their computer skills through high school and university courses, although nearly half the group had used computers in their homes (Smith and Furst-Bowe, 1992). Additional studies have found that factors such as gender, age, experience and interest in computers can affect students' attitudes toward and use of computers (Morahan-Martin, 1992). The findings of these and other studies suggests that additional research needs to be done to identify factors that could influence students' abilities to use computers. Few colleges or universities have any type of computer skills requirement or any kind of system to ensure that all students enter the university or graduate from the university with the computer competencies necessary to be successful in the world of work.

Methodology: In view of these findings, our research addressed four main objectives:

1) to identify computer competencies of freshman students
2) to identify computer competencies program directors (major advisors) expected of their students
3) to identify computer competencies necessary for entering the workforce as viewed by graduates, and
4) to identify computer competencies necessary for entering the workforce as viewed by employers.

These four groups were surveyed with the attached instruments (B through E) and a cover letter (A), which addressed individual program directors, graduates, and employers, but not students, by name. The survey instrument asked the same competency questions of each group, although demographics and contextual questions differed. Students were asked to identify their present computer skills and how they were obtained, while program directors were asked what competencies their students should have, when, and how important they would be to future employment. Graduates and their employers were asked about computer competencies for new employees, identified according to type and size of organization.

Computer competencies were grouped under Basic Computer Skills (12 items), Word Processing Skills (7 items), Spreadsheet Skills (6 items), Database Skills (7 items), Graphics/Multimedia Skills (6 items), and Information Retrieval/ Telecommunications (5 items). Respondents answered "Yes," or "No," or "Unsure" about whether the skill was important to them, to their students, or to their employees.

Freshman students in eight sections of entry-level English composition were surveyed, with a resulting n=154, of whom 55% (84) were female and 45% (70) male. Fifty-five percent (55%) were freshmen (86), 20% sophomores, 14% juniors, and 10% seniors. Business, education, industrial technology, and hospitality were the most heavily-represented majors, accounting for 57% of those surveyed, with retail and human services majors next (15%), and the rest distributed among others ranging from art to dietetics to applied mathematics. Thirty-seven percent (37%) owned computers, half IBM and the rest MAC's or others, while 63% did not.

Twenty-seven (27) program directors, or major advisers, from thirty-two undergraduate programs responded to the survey. With the exception of art, which includes industrial and graphic design, and applied mathematics, which includes computer science, all of the programs represented are in the fields of home economics and hospitality, including retail merchandising and manufacture, education and human services, including psychology and vocational rehabilitation, and industry and technology, including engineering and business administration. Fifty randomly-sampled graduates from May, 1991, were surveyed. The twenty-eight (28) respondents represented primarily business and educational institutions, with production, hospitality, retails, human services, and others included in smaller numbers. Twelve (12) of the 28 represented business with less than 100 employees, eight represented those with 100 to 500, and the remaining nine organizations with over 500 employees, with two over 10,000. Of these graduates twenty -two (of 28) used IBMs alone or with others. The UW-Stout Placement Office provided the names of the last 100 campus recruiters entered into the computer database of approximately 1925 employers who have recruited from the campus. Of these 72 responded, 16 from business, 12 from hospitality, eight from retail, and the rest distributed among other industries, including manufacturing and food-related. More than one-third (25) of the organizations they represented employed fewer than 100 people, nearly another one-third (21) between 100 and 1000; sixteen (16) 1000 to 10,000, and 8 more than 10,000 employees. Of these organizations, 83% used IBMs alone or with others.

Core Computer Competencies

Table 1 presents the mean percent of "yes" responses organized by the main categories of competencies to enable comparisons across categories and samples:

Table 1 Mean Percent of "Yes" Responses for Skill Categories

RESULTS

This research addressed four main objectives:

a. Identify computer competencies of freshmen students at UW-Stout
b. Identify computer competency expectations of students by undergraduate Program Directors
c. Identify computer competency expectations of graduates upon entering the work force as viewed by alumni
d. Identify computer competency expectations of graduates upon entering the work force as viewed by employers

These objectives were addressed within survey instruments which contained demographic items for each of the four research samples as well as 43 computer competencies presented within categories.

Core Computer Competencies

Table 1 presents the mean percent of "Yes" responses for each of the survey samples, organized by the main categories of competencies within the instrument. Since the number of items within categories of competencies varied, the mean percent of "Yes" responses enabled comparisons across categories and samples.

Table 1

Mean Percent of "Yes" Responses for Skill Categories

	STUDENTS (N=157)	PROG. DIR. (N=27)	ALUMNI (N=28)	EMPLOYERS (N=72)
BASIC SKILLS (1-12)	54	73	48	48
WORD PROCESSING (13-19)	71	69	53	49
SPREADSHEET (20-25)	38	62	40	54
DATABASE (26-32)	22	58	28	37
GRAPHICS/MULTIMEDIA (33-38)	34	65	35	28
INFO. RETRIEVAL/TELECOM. (39-43)	39	66	25	22
TOTAL OVERALL ITEMS (1-45)	44	65	40	40

Students: Of the seven groups of competencies, those most students (71%) felt comfortable with involved word processing, with slightly more than half (54%) so with basic skills, and fewer with spreadsheets, database, graphics, and information retrieval.

Program directors, graduates, and employers: At least two of these three groups agreed by more than 50% "yes" on the importance of word processing and spreadsheets. Of the three groups, program directors believed most strongly in the importance of computer skills, more than half responding "yes" for all seven categories. Graduates and employers were affirmative about basic skills; graduates relied on word processing, whereas their employers identified spreadsheets.

Table 2 provides a more detailed account of respondents' skills and attitudes by presenting the percent of "yes" responses for all forty-three competencies. Analysis of these results provides insights into computer competencies to be addressed within our curriculum.

Students: More than half of these freshman students possessed the basic skills necessary to turn on a computer, use Windows and, to a lesser extent, DOS or MAC operation systems, start a software program, copy files, and manage a hard drive. High percentages were competent in all wordprocessing areas save for resumes and brochures. Only 55%, however, were able to enter data into an existing spreadsheet, the only item in the spreadsheet group which more than half felt competent. Less than half the group felt competent in database, and only 8% were able to do database programming. In graphics/multimedia 51% could draw simple shapes, but fewer than half the group could accomplish the remaining items. Eighty-two percent could use electronic databases (ERIC, INFOTRAC), the only item above half in this section.

Program Directors, Graduates, and Employers: Items on which more than half of two of these three groups agreed were important were:

Basic skills: turn on/off, use Windows and DOS (but not MAC), format a floppy disk, start a program, copy files, and teach yourself a new program.
Word processing: business letters, proposals, resumes, and outlines.
Spreadsheet: entering data, creating a new spreadsheet, functions, and formulas.
Graphics/multimedia: none, although 82% of program directors identified creating graphs and charts.
Information retrieval/telecommunications: e-mail.

Table 2 Mean Percent of "Yes" Responses for Computer Skills

Table 2 provides a more detailed account of respondents' skills and attitudes by presenting the percent of "Yes" responses for each of the samples for all 43 competencies. Analysis of these results provides insights into computer competencies to be addressed within the curriculum of the university. Although students' self reported skills are generally consistent with alumni and employers' expectations, careful attention is advised to the Program Directors' attitudes regarding skills they believe their students need to not only enter the work force but progress in the work force.

Table 2

Mean Percent of "Yes" Responses for Computer Skills

	STUDENTS (N=157)	PROG. DIR. (N=27)	ALUMNI (N=28)	EMPLOYERS (N=72)
BASIC SKILLS (1-12)				
1. Turn on Computer	99	100	93	89
2. Use Windows operating system	78	63	50	59
3. Use DOS commands	52	67	50	59
4. Use MacIntosh operating system	61	44	15	15
5. Format floppy disk	55	82	50	54
6. Start software program	54	96	68	56
7. Copy files	63	96	61	63
8. Manage hard drive (folders/directories)	63	78	64	53
9. Set up new personal computer from box	32	41	21	16
10. Install new software on computer	34	74	32	30
11. Teach yourself new software program	48	96	68	67
12. Set up computer network	12	32	7	00
WORD PROCESSING (13-19)				
13. A business letter	92	89	86	83
14. A research paper or proposal	95	96	57	60
15. A resume	75	85	71	49
16. A mail merge (form letters, mail labels)	47	44	32	31
17. An outline	87	82	54	68
18. A newsletter	72	48	39	27
19. A brochure or complex publication	31	37	32	14

SPREADSHEET (20-25)				
20. Enter data into existing spreadsheet	55	74	50	77
21. Create new spreadsheet	48	63	43	61
22. Create functions/formulas for spreadsht	35	59	43	51
23. Create charts/graphs from spreadsht data	38	70	43	49
24. Create macros for spreadsheet	20	44	21	29
25. Use database functions (e.g. sort, query)	31	63	39	54
DATABASE (26-32)				
26. Enter data into existing database	41	74	50	71
27. Create new database	26	70	29	37
28. Create functions/formulas for database	23	59	18	30
29. Create a database report	21	59	29	39
30. Sort and query a database	21	59	39	41
31. Use indexes/views to link databases	12	52	21	22
32. Do database programming	08	27	11	12
GRAPHICS/MULTIMEDIA (33-38)				
33. Draw simple shapes and objects	51	78	46	40
34. Use clip art	40	52	39	27
35. Draw complex objects/illustrations	26	63	21	10
36. Create charts, graphs, diagrams, flowchts	38	82	50	40
37. Create presentation materials, overheads	30	67	43	39
38. Create multimedia presentation	20	50	11	10
INFO. RETRIEVAL/TELECOM. (39-43)				
39. Send and receive electronic mail	47	78	44	61
40. Use electronic databases (PUBCAT, ERIC)	82	85	29	13
41. Use an electronic bulletin board	23	63	32	12
42. Locate and retrieve info over Internet	35	70	18	15
43. Set up a computer conference	09	33	4	3

As further perspective on desireable computer competencies, Table 3 presents the mean percent of "Yes" responses from alumni, indicating skills needed in their work, organized by skill categories on the questionnaire, for different types of organizations in which those alumni were employed. Faculty and Program Directors structuring curriculum targeted for particular types of employment might find these results of interest. It should be noted that for some types of organizations the sample sizes were very small.

The top five computer skills as identified by students were:
1) turn on/off a computer, monitor, and printer (99%)
2) produce a research paper on a wordprocessor (95%)
3) produce a business letter on a wordprocessor (92%)
4) produce an outline on a wordprocessor (87%)
5) use electronic databases (82%)

Program directors identified three of these (turn on/off and wordprocess a research paper and business letter) among their top five, although with less unanimity about the need to produce a business letter (89%). Instead of databases and outlines, however, they chose "start a software program" (100%) and "Use DOS commands" (96%). Graduates identified those two as well, in addition to (turn on/off), along with the research paper and business letter. Employers identified "turn on/off" and business letters as well, agreed with students on the importance of outlines (68%), and added entering data on a spreadsheet (77%) and into an existing database (71%).

As a further perspective on desirable computer competencies, Tables 3 and 4 present the mean percent of "yes" responses from graduates and employers on skills needed in different types of organizations. Those structuring curriculum for particular types of employment might find these of interest, although for some types of organizations sample sizes were very small.

Importance of Programming Skills

In addition to the 43 specific competencies addressed in the questionnaire, item 44 asked whether students and graduates had programming or other computer skills, and whether program directors and employers though they were important. Although 32% of the students reported having programming skills, only 21 of graduates, 19% of program directors, and 11% of employers found them important.

Importance of Computer Skills for Employment

Sixty-eight percent (68%) of students saw computer skills as "essential" (41%) or "very important" (27%) to obtaining employment, whereas 23% saw them as "somewhat important," two percent (2%) as "not important," while seven percent (7%) were unsure. Almost two-thirds (61%) saw high school alone or with others as the place to get computer skills, while 24% placed faith in word (11%) or home (13%). Only nine percent (9%) saw the university as such a place. Program directors saw the university as providing most of these skills (90%), on the other hand, with high school (70%) or work (70%) being secondary sites for such acquisition. Only 38% of these directors thought students needed computer skills upon entry, although another third (31%) thought they would be needed at midpoint in a students' career, and 31% though they would be needed upon graduation.

Table 3

Mean Percent of Alumni "Yes" Responses for Skill Categories for Different Types of Organizations

	Basic Skills (1-12)	Word Proc. (13-19)	Spreadsheet (20-25)	Database (26-32)	Graphics (33-38)	Info/Tech (39-43)
Business	54	73	48	48	60	35
Education	46	50	17	17	39	20
Industrial Tech	66	57	78	52	39	33
Retail	25	43	00	00	00	40
Hospitality	29	71	17	00	00	30
Human Services	42	57	33	43	28	40
Art/Design	33	29	00	00	00	00
Food/Dietetics	83	43	100	100	50	00
Other	31	48	61	19	28	00

Table 4 presents the mean percent of "Yes" responses from employers, indicating skills needed in their employees, organized by skill categories on the questionnaire, for different types of organizations in which those employees worked. Faculty and Program Directors structuring curriculum targeted for particular types of employment might find these results of interest. It should be noted that for some types of organizations the sample sizes were very small.

Table 4

Mean Percent of Employers "Yes" Responses for Skill Categories for Different Types of Organizations

	Basic Skills (1-12)	Word Proc. (13-19)	Spreadsheet (20-25)	Database (26-32)	Graphics (33-38)	Info/Tech (39-43)
Business	56	44	61	43	35	14
Education	92	86	83	71	100	100
Industrial Tech	57	53	97	50	44	20
Retail	32	48	14	06	12	26
Hospitality	38	51	58	40	21	13
Human Services	45	46	21	36	04	20
Art/Design						
Food/Dietetics	37	41	38	33	24	43
Other	64	56	65	43	35	23
Multiple Response	00	57	17	14	00	20

Conclusions

1) Although students' self-reported skills are generally consistent with graduates and employers' expectations, programs directors and classroom teachers should pay attention to skills they believe students need not only to enter but to progress in the work force. They must pay attention as well to integrating practice in these skills into the curriculum in order to refine and upgrade them.
2) Seventy-four percent (74%) of major advisors said computer courses were included in their curriculum, whereas 93% had said the such skills were either essential or very important. The lack of consensus about whether, where, and how these skills would be attained reflect differences among majors and their specific expectations, but point to the need at least for consensus on basic skills and the university's ability to provide them.
3) Since freshman students at least do not see the university as a place in which computer skills will necessarily be obtained, efforts to make basic skills training and computer accessibility known should be intensified during freshman orientation and during the freshman year. It is to be inferred that the high degree of familiarity students have with electronic databases is due to library orientation in beginning classes.
4) Training for initial familiarity with spreadsheets and database should be more focused on, whereas instructors using wordprocessing, with which students are already familiar, should focus on higher level uses such as revisions and block moves, columns, and desktop publishing.
   Efforts should be made to identify the thirty-percent of students who are not familiar and tailor training specifically to their needs.
5) Instructors, advisers, and curriculum planners need to continue to encourage students to develop computer skills as means of advancing in their careers, and not necessarily as tickets to entry level positions. Emphasis should be placed on computer skills as communications and managerial strategies.
6) Information on computer competencies such as that contained in this study should be disseminated among faculty and staff; instructional and curriculum development efforts should focus on ways in which these competencies can be infused into the curriculum.
7) Means should be sought to include computer competencies into the skills and abilities required for general education; program assessment in general education and in the disciplines should address these competencies in their review.
8) Surveys of graduates and employers should continue as part of the institution's assessment measures in order to document needs and improvements.

TQM Campus Dynamics

Although in some ways our project came to look like an ordinary research exercise, the TQM structure and tools allowed us to:

1) see our problem as one of design, which lent a necessary perspective to data gathering, defining aims, and setting objectives;
2) see students, advisers, and employers as different groups of customers or cli-

ents whose needs could be identified and assessed;
3) use new tools for problem identification and solution--different charts and diagrams, especially fishbone and cause-effect charting;
4) create more effective group dynamics with our sponsor, leader, and facilitator;
5) enjoy the security of knowing the group, representing different perspectives and areas of involvement, shared a high level of commitment to the goals of the project and trust in one another;
6) address issues of credibility and our ability to follow through to implementation forthrightly because of our implementation charge;
7) see ourselves as empowered because of our ability to make recommendations for disseminating and following up upon our preliminary identification of competencies.

Our outcomes were not as ambitious as those we had first envisioned. Although we were able in the five months we were together to identify preliminary competencies, it would need to be left to another team to identify the desired levels of competence, or outcomes, we should look for in our curriculum. These levels would then become the benchmark against which ensuing measurements would be judged. Roy Bauer, a 3M executive whose Rochester-based TQM team was awarded a Baldridge, and who now sits on the Baldridge assessment team, addressed our group last month. He said that the important thing about quality and TQM tools is to keep them flexible, and that the only element which all quality projects necessarily must share is attention to the customer. We felt that TQM group 3 remained true to this ideal in both its process and its product, and that perhaps some of our most satisfied customers' at the end were the team members themselves.

Sources Consulted

TQM: Team Training for Process Improvement, workshop and materials by Casey Collett and Joan DeMott.

Computing and Telecommunications Technology Task Force Proposal. University of Wisconsin-Stout, October, 1993.

Harrington, J.L. (1990, May). Changes in the computer background of incoming freshmen. Collegiate Microcomputer. 8, 147-154.

Hirschbuhl, J. and Faseyitan, S. (1994). Faculty uses of computers: fears, facts and perceptions. Technological Horizons in Education Journal. 21(8), 64-65.

Lloyd. B.H. and Gressard, C. (1984). The effects of sex, age, and computer experience on computer attitudes. AEDS Journal, 67-77.

Morahan-Martin, J. (1992). Gender differences in computer experience, skills and attitudes among incoming college students. Collegiate Microcomputer. 10, 1-7.

Secretary's Commission on Achieving Necessary Skills. (1991). What work requires of schools: a SCANS report for American 2000. Washington DC: U.S. Department of Labor.

Smith, M. and Furst-Bowe, J. (1992). An assessment of computer skills of incoming freshmen at two University of Wisconsin campuses. Paper presented at the Association for Educational Communications and Technology 1993 annual conference in New Orleans, LA.

Strickland, J. (1989, April). How the student adapts to computers. Computers and Composition. 7-22.

Sullivan, P. (1989). What computer experience to expect of college students entering a computer classroom. Journal of Technical Writing and Communications. 19. 53-68.

Ethical Issues for Information Systems Professionals

Thomas A. Pollack
A.J. Palumbo School of Business Administration
Duquesne University
Pittsburgh PA 15282
412-396-1639
pollack@duq2.cc.duq.edu

Abstract

It has become increasingly apparent to educators and professionals alike that, in addition to the technical issues of information systems, social, ethical, and professional implications are also important for information systems professionals. Computers and information systems have a significant impact on the world in which we live, and it is extremely important for information systems professionals to gain an understanding of social and ethical implications.

This paper discusses prevalent ethical and social issues that should be woven into an information systems curriculum. Topics such as privacy and accessibility, software piracy and property rights, codes of professional conduct, hacking, and liability will be addressed. The paper will also briefly examine the curriculum implementation and presentation strategies for the above topics.

Introduction

A significant component of the mission statement for students of Duquesne University reads:

> ...To examine the moral and ethical foundations of their thought and action, and to develop their personal values and ethical commitment.

The Ethics Committee of the A.J. Palumbo School of Business Administration was formed in 1990. To support the University Mission Statement, the committee formulated a resolution which was unanimously adopted by the faculty in September 1990 which reads:

> It is the intent of the faculty that a consideration of ethics in business be made an integral part of the graduate and undergraduate curricula in the School of Business.

In October of 1991, the A.J. Palumbo School of Business Administration embarked on a complete revision process for its undergraduate curriculum. Through interaction and feedback from students, alumni, members of advisory boards, and the business community, the following fundamental themes emerged as common elements to be integrated across the curriculum: (1) Integration of business disciplines, (2) Human resource issues, (3) Ethics, (4) Total quality management, and (5) Global Perspective.

In May of 1993, the Graduate School of Business Administration initiated a similar curriculum revision process. Once again, much interaction with identified "customers" presented common themes for graduate study. The common cross-cutting themes include: (1) Ethics, (2) Total quality management, (3) Global perspective, and (4) Management of technology.

The author presents the above background material to demonstrate the commitment being expressed by Duquesne's School of Business for inclusion of ethical components within the curricula. This is in keeping with a growing momentum that states "ethics is good business." Indeed, ethics integration is an area that each of our concentrations continues to explore and develop. The intent of this paper is to develop a rationale and framework for ethical issues which should be woven into an information systems curriculum.

Ethical Decision Making

Dejoie, Fowler, and Paradice (1991) introduce their book of readings on ethics by differentiating the terms moral and ethical. The term moral is generally associated with conformity with generally accepted standards of goodness or rightness in conduct or character. The term ethical is associated with conformity with an elaborated, ideal code of moral principles, such as those often published by professional organizations. Taylor (1975) concludes that ethics may be defined as "inquiry into the nature and grounds of morality where the term morality is taken to mean moral judgements, standards, and rules of conduct." Therefore, business ethics involves moral judgements, standards and rules of conduct in situations involving business decisions.

It is difficult to reach agreement on a suitable definition of ethics. Some, for instance, want to associate that which is legal as being ethical. However, many researchers and professionals feel that organizations also have social responsibility. Social responsibility is really ethics at the organizational level, since it refers to the obligation that an organization has to make choices and to take actions that will contribute to the good of society as well as the good of the organization. Social responsibility requires a voluntary response from an organization, above and beyond what is specified by law (Richardson, 1991).

The notion of social responsibility is in no way an attempt to minimize the legal impact on ethical behavior. Local, state, and federal government regulatory acts have had significant impact on business ethics and the conduct of many business activities, even to the point of being perceived as intruders into private business matters and overregulators of the corporate sector (Madsen and Shafritz, 1990). Similar controversy surrounds the question of whether ethics can be taught at the collegiate or professional level of an individual's career.

For purposes of discussion in this paper the author will use the definitional assumption proposed by Kallman and Grillo (1993). They state that "...ethical principles are ideas of behavior that are commonly acceptable to society. Using ethical principles as a basis for decision making prevents us from relying only on intuition or personal preference. ...Ethics is the practice of making a principled choice between right and wrong." Managers are further influenced in decision making by family, peer, and religious influences as well as past experiences and unique personal value systems (Richardson). It is further assumed, as stated by Pastore (1993) that "very little is black and white... but instead gray areas which lack clear cut answers are those that cause sleepless nights and create ethical dilemmas."

Madsen and Shafritz (1990) contend that business ethics is an example of applied ethics; it involves taking the pure theories, principles and concepts of formal ethics and applying them to the world of work. They further contend that business ethics can be divided into two separate areas each with its own set of issues, problems, and dilemmas. The first area is referred to as managerial mischief. It concerns the illegal, unethical or questionable practices of managers or organizations, their causes and their possible remedies. The second is referred to as moral mazes in management. It is concerned with the numerous ethical questions that managers must confront as part of their daily business decision making.

O'Brien (1994) proposes two fundamental questions which business ethics must address; (1) How do business professionals come to know what is right in carrying out business tasks? (2) How can they them do what is right once they have recognized it?

Ethics In Information Systems

The previous section of this paper provides a basis for the IS importance of including ethical components in the instructional process for young IS professionals. This section will examine some of the specific areas of concern for information systems professionals.

Peter Drucker (1981) indicated that business ethics was rapidly becoming the "in" subject. In a more recent article, Pastore states, "moral convictions aside, careful attention to ethics is also sound business practice, particularly for firms handling sensitive data. Vitell and Davis (1990) report that the functional area of business that currently may be experiencing the most ethical problems and conflicts is information systems. Part of the reason is the fact that other business professions such as accounting, finance, and marketing are well-established and have developed ethical codes over many years. The information systems field is relatively young and does not have the same historical precedent. The author adds to this the fact that the information systems field has experienced dynamic change and is ever evolving.

In studies completed by Oz (1992) and (1993), four major professional codes of conduct are analyzed. Earlier in the paper the statement was made that ethical principles are ideas of behavior that are commonly acceptable to society. Oz states that activities of computer professionals have a tremendous impact on society, but professionals in the information technology field do not have a single agreed upon code of conduct. He analytically assesses the codes of conduct espoused by the four largest computer professional organizations in the U.S. Included are the Data Processing Management Association (DPMA) the Institute for the Certification of Computer Professionals (ICCP), the Association for Computing Machinery (ACM), and the Information Technology Association of America (ITAA).

Oz concludes that the relationship between computer professionals and the public is similar to most other professional relationships in terms of knowledge and reliance. Even though the field is changing and technological advances raise new ethical issues, Oz encourages the development of one, coherent code of professional conduct. It is felt that a single code would help in fostering better public recognition of information systems professionals, and greater commitment of the professionals to society, employers, clients, and colleagues.

Karen Forcht and Anne Myong reported that more than half of the 300 business students polled in a recent survey at James Madison University admitted to using computers for "unethical means" including software piracy and hacking into off-limits systems (Slater, 1991). This finding has been confirmed at other campuses and should cause great concern.

Glenn Rifkin (1991) also writing in the same Computerworld issue attributes the following statement to Donn B. Parker, an international authority on ethics, computer crime, and privacy:

> Ethics requires intelligence. You've got to be smart, especially in the area of human relations, and that's difficult for IS people to do. They are more skilled at interacting with the computer than with people.

Rifkin (1991) also reports that the subject of ethics doesn't often arise among IS professionals. Observers say the main reasons for disinterest appears to be the nature of IS people and their tasks, fear of losing one's job and a widespread belief that an ethics policy is best handled by general management.

Susan Athey (1993) concludes that a need exists to expose future IS professionals to the different situations that develop as a result of our increasingly electronic world. Athey opines that problems and possibilities exist which no one had to worry about 30 years ago. Kidder (1992) promotes a similar view. He writes:

> ...our technology has leveraged our ethics in ways that we never saw in the past. ...we spend tremendous amounts of effort in our educational system teaching about the nature of technology - and virtually no time at all talking about the moral and ethical consequences of that technology.

However, all is not negative with regard to studies and conclusions involving IS professionals. Vitell and Davis (1990) actually conclude that MIS professionals feel there are many opportunities for unethical behavior, but they also believe that MIS managers within their companies are unlikely to engage in unethical behavior. Another finding of the study was that MIS professionals have a very strong sense of social responsibility. They believe that the corporation has a responsibility to society beyond its responsibility to its shareholders. Vitell and Davis also conclude that there is a positive relationship between success and ethical behavior.

Ethics In Information Systems: Specific Issues

The overwhelming consensus of research studies indicates that upcoming IS professionals should be made aware of ethical and societal issues related to the discipline. Technical expertise alone is not enough. Technology has had a dramatic impact on our world and our society, and items which were "non-issues" only a few years ago now hold the prospect of creating ethical dilemmas.

There is a degree of concurrence among researchers and IS professionals as to which issues are important in developing a sense of ethical awareness. The author has considered a number of reference sources and has attempted to arrive at a general consensus of topics which should be addressed from an ethical perspective. Students in information systems programs should be exposed to four general areas of ethical concern. Among them, according to Dejoie et al. (1991) are (1) the information systems role in the decision-making process, (2) the perceived opportunity (or lack thereof) to commit ethical or unethical activities. (3) issues of ownership and responsibility for information systems, and (4) the "shrinking world" issues caused by information systems and telecommunications systems. Mason (1986) summarized four key ethical issues with the acronym PAPA. The issues include Privacy, Accuracy, Property, and Accessibility.

Sources such as O'Brien (1994), Kallman and Grillo (1993), Ermann et al. (1990), Dejoie et al. (1991), Oz (1993) and Pastore (1993) repeatedly touch upon many of the same specific areas of ethical concern. The author will enumerate some of the more prevalent issues and provide a brief description for each area of concern.

1. Data integrity has to do with the accuracy of data at its source and the measures taken to ensure its accuracy as it wends its way through the labyrinth of distributed systems. Sometimes staff auditors are employed in this process.

2. Data security has to do with granting flexible access while at the same time shielding data from unauthorized users.

3. Internal IS ethics must deal with the pressures associated with strict budget limits and unreasonable delivery deadlines and the tempting notion that developers might compromise professional ethics by cutting corners.

4. Right to privacy has to do with such issues as electronic mail and system monitoring (Bjerklie 1993) and (Nash 1991) and corporate policies which help to manage and inform on such issues.

5. Computer crime encompasses such issues as money theft, service theft, and software theft. The most controversial issue is software piracy which is an issue of enforcing copyright laws, but practically speaking, is difficult to enforce.

6. Data alteration involves making illegal changes to data.

7. Computer viruses involves one of the most destructive examples of computer tampering. Often, viruses destroy the contents of memory, hard disks, and other storage devices. Vaccine programs can help diagnose and remove computer viruses.

8. Hacking is the obsessive use of computers, or the unauthorized access and use of computer systems. The real problem arises when a hacker is guilty of "electronic breaking and entering."

In addition to the long list of ethical issues, many of which laws and regulations attempt to control, other humanistic issues also warrant our attention. Among this list of concerns are automation and employment, the negative impact computers have had on the individuality of the person, health issues, vendor relationships and conflicts of interest.

Pedagogy

Each of the issues, in the previous section warrants the attention of students in information systems. The issues should be methodically integrated into a program of studies. Exactly how to deliver ethics instruction in information systems courses remains a topic for discussion. Some will question whether ethics instruction at the collegiate level is effective, especially when peer and job-related pressures are considered. Few, however, will question the logic of exposing students to issues which are in the "gray area" and allowing them to make decisions.

At Duquesne University, several programs are in place to assist faculty with coverage of ethical issues. An alumnus of the School, Eugene Beard has endowed the Beard Center for Leadership in Ethics. One of the ongoing initiatives of the Beard Center is a Distinguished Speaker Series. The speaker series is sponsored by another distinguished alumnus, John W. McGonigle. Students are generally asked to attend the Speaker Series or watch the video tape and present an abstract of the discussion topic. Follow-up discussions are frequently conducted.

Recently, the faculty of the School completed a workshop session which addressed the issue of discipline-specific integration of ethical issues. Since the goal of the School is to include ethical issues as one of the common threads of our curricula, systematic integration of issues is important. Most faculty have been addressing the issue of ethics by using cases which encourage class interaction. The cases normally address the afore-mentioned "gray areas" of decision-making. Cooperative learning strategies are employed to involve and empower students in ethical decision-making.

Another effective teaching tool employed by some faculty is a discussion of the code of conduct published by discipline-specific professional organizations. This strategy has proven to provide an effective lesson for ethical behavior as well as career expectations. Often, this strategy is coupled with a guest appearance on the part of a practitioner.

Articles by Cohen and Cornwell and also by Couger (Dejoie et al. 1991) address the area of pedagogy. Cohen and Cornwell report that ethics are best taught and learned through integration into the curriculum. The recommended approach is to pose ethical questions to students and discuss their responses.

Couger concludes that two objectives can enhance student reaction to ethics instruction. The first is instructor emphasis on the importance of the subject. The second is to motivate students to incorporate a code of ethics into their behavior by personalizing issues discussed in class.

Summary

Researchers and professionals alike encourage the exposure of IS students to ethical issues. Lacking a single code of conduct, those in the computing and information systems profession do not enjoy the luxury of other longer-established disciplines. Add to this the fact that technology has undergone such dramatic changes over the past 15 years and the opportunity for indecisive action may prevail. Advancements in personal computers, networks, and distributed systems, for instance, have had a significant impact on the privacy issue.

Common ethical issues for information systems have been identified in the literature, therefore, the question is not one of which topics should be discussed. The question is more focused on how to effectively develop a conscious sense of awareness and influence the future actions of tomorrow's professionals. The theme that "ethics is good business" is growing in momentum, and educators in the information systems field must do their part to see that our future information systems professionals "do the right thing, right."

References

Athey, S. (1993). A Comparison of Experts' and High Tech Students' Ethical Beliefs in Computer-Related Situations. Journal of Business Ethics, 12, 359-370.

Bjerklie, D. (1993). E-Mail: The Boss is Watching. Technology Review, April, 14-15.

Dejoie, R. Fowler, G., and Paradice, D. (eds.). (1991). Ethical Issues in Information Systems. Boston: Boyd & Fraser Publishing Company.

Drucker, P.F. (1981). What is Business Ethics? Across the Board, October, 22-32.

Ermann, M.D., Williams, M.B. and Gutierrez, C. (eds.). (1990). Computers, Ethics & Society. Oxford: Oxford University Press.

Kallman, E.A. and Grillo J.P. (1993). Ethical Decision Making and Information Technology: An Introduction with Cases. New York: Mitchell McGraw-Hill.

Kidder, R.M. (1992). Ethics: A Matter of Survival. The Futurist, March-April, 10-12.

Madsen, P., and Shafritz, J.M. (eds.) (1990). Essentials of Business Ethics. New York: Meridian.

Mason, R. (1986). Four Ethical Issues of the Information Age. MIS Quarterly, March, Vol, 10.

Nash, J. (1991). Technology Raises Many New Ethics Questions. Computerworld, October 14, 89.

O'Brien, J.A. (1994). Introduction to Information Systems (7th ed.). Burr Ridge, Illinois: Irwin.

Oz, E. (1992) Ethical Standards for Information Systems Professionals: A Case for a Unified Code. MIS Quarterly, December, 423-433.

Oz, E. (1993). Ethical Standards for Computer Professionals: A Comparative Analysis of Four Major Codes. Journal of Business Ethics, 12, 709-726.

Pastore, R. (1993). Ethical Gray Matters. CIO, 6(7), 58-62.

Richardson, J.E. (ed.) (1991). Business Ethics 91/92. Guilford, Connecticut: Dushkin Publishing Group, Inc.

Rifkin, G. (1991). Meet an IS Ethics Crusader and His Long and Lonely Cause. Computerworld, October 14, 90.

Rifkin, G. (1991). The Ethics Gap. Computerworld, October 14, 83-85.

Slater, D. (1991). New Crop of IS Pros on Shaky Ground. Computerworld, October 14, 90.

Taylor, P.W. (1975). Principles of Ethics; An Introduction. Encino, California: Dickson Publishing Company.

Vitell, S.J. and Davis, D.L. (1990). Ethical Beliefs of MIS Professionals: The Frequency and Opportunity for Unethical Behavior. Journal of Business Ethics, 9, 63-70.

Listserv Discussion Groups on Internet/Bitnet

James W. Reese, Ph.D.
Associate Professor of Economics
University of South Carolina, Spartanburg
800 University Way
Spartanburg SC 29303
803-472-8524
r505040@univscvm.csd.scarolina.edu (Internet)
r505040@univscvm (Bitnet)
Editor: japan@pucc.princeton.edu
e-europe@pucc.princeton.edu
quality@pucc.princeton.edu

Introduction

Over the past five years, the number of Internet/Bitnet mailing lists has exploded. There are presently over a thousand lists on practically any subject, and they exist with a wide variation in features and options. This paper is a brief overview of the creation and operation a Listserv mailing list.

Listserv discussion groups are individuals with a common interest which are linked together worldwide by e-mail. "Listserv" refers to the mailing list software which automates most of the activities of the list. A "list" refers to the list of subscribers of a discussion group. Often the group will be called a SIG (special interest group), a network, or a mailing list. Listserv groups are found almost exclusively on academic sites, and it is the dominant mailing list software.

Lists are located at a listserv "host" university. A host is usually a university which has both Internet and Bitnet node addresses. Many hosts will be major listserv sites with fifty or more lists in operation. Lists are created and managed by owners/editors, the majority of which are not faculty of the listserv host university. The actual location of the editor(s) is immaterial. The only requirement is electronic access.

For example, the author is a faculty member of a regional campus which does not have an Internet/Bitnet node. He connects to the mainframe computer on the main campus by the statewide university telephone system. From that site, he operates three discussion groups located at Princeton University.

Mailing Lists

A simple mailing list can be created by using the "cc" option of an e-mail system. For a small group of individuals with low turnover, this procedure is more than adequate. E-mail postings and the addition/removal of individuals is done by the person mailing the documents.

For a larger group of public subscribers, this procedure is unworkable. The list editor would spend most of his/her time handling subscription requests. This is when automated mailing list software, such as Listserv, is needed. Over the past eight years, Listserv has evolved to the point of managing most of the mechanics of a mailing list. This includes the addition and removal of subscribers, deleting subscribers with expired e-mail addresses, as well as sophisticated file storage and search queries.

In order to become an "owner/editor" of a mailing list, one must master: the mechanics of a pc and its communications software (using just a terminal of a mainframe would be inadequate); the communications procedure between the pc, the linking mainframe, and the Internet/Bitnet host; the mailing system of the linking mainframe and the Listserv host; and the basic Listserv commands for subscribers and editors. Some of the mainframe software is as primitive and unfriendly as it was twenty years ago; but this is offset by the recent advancement in pc communications software.

Types of Mailing Lists

Mailing lists can take almost any form. On one extreme, it can assume the form of a peer-reviewed electronic journal published quarterly to members of an association. Although no fees can be charged for subscription to a Listserv mailing list, subscription can be open to only those dues paying members of an association.

At the other extreme, the mailing list can be completely open to the public and unmoderated. However, the anarchy which follows drives out most serious contributors.

If the objective is to create a mailing list devoted to serious discussion of a subject, the list must be either moderated by the editor or subscription limited to invitation only (i.e. non-public). Otherwise, diatribes in the form of e-mail "flame wars" result. Monitoring any large unmoderated USENET newsgroup will quickly confirm this.

Mailing List Activities

There are two primary activities performed by a Listserv mailing list. Its primary function is the mass distribution of e-mail messages to subscribers; the secondary one is storage and distribution of files.

1. Message Distribution

E-mail discussion is the reason for the explosion in mailing lists. With the rapid expansion of Internet access worldwide, geographical distance becomes largely immaterial. Questions posted in New York can receive a reply from Australia in a matter of minutes; followed by a rebuttal from Warsaw shortly thereafter.

For example, if one has a question about monetary policy in Poland, sending one e-mail message to E-EUROPE@PUCC will relay it in a matter of minutes to 1300+ subscribers in 50 countries. However, since E-EUROPE is a moderated list, the message would first be reviewed by the list editor before posting to the subscribers.

The latest versions of Listserv, 1.7f and above, allow much greater flexibility in message distribution to subscribers. Subscribers can receive messages in four formats: 1) full text of all messages when they are posted to the list; 2) an daily or weekly digest of the full text of all messages; 3) a daily or weekly index of the messages with the headers (subject, from, to, date), with the ability to retrieve only those messages of interest; or 4) an editor created "digest" posted whenever.

There is also a topics option. If enabled, subscribers can control which "topics" they want to receive. For example, with the NEW-LIST@NDSUVM1, you send one of the commands below to listserv@vm1.nodak.edu or on BITNET to LISTSERV@NDSUVM1 in the BODY of the mail:

```
set new-list topics=change,new
        to receive only changes and new list announcements
set new-list topics=search
        to receive only list searches
set new-list topics=all
        to receive all items (this is the default when you subscribe).
```

2. Message Index Example

On March 24, 1994, the JAPAN@PUCC added the index option for e-mail. Those subscribers who selected index, received the following message the next day. Included in the message is the header information and a document number (00640) and size (15 lines):

```
Sender:  Japanese Business and Economics Network <JAPAN@PUCC.BITNET>
From:    Automatic digest processor <LISTSERV@PUCC.PRINCETON.EDU>
Subject: JAPAN Index - <first ever> to 24 Mar 1994
To:      Recipients of JAPAN indexes <JAPAN@PUCC.PRINCETON.EDU>


Index Date  Size Poster and subject
----- ----  ---- ------------------
00640 03/24   15 From:   "James W. Reese" <R505040@UNIVSCVM.BITNET>
                 Subject: New Listserv Options for JAPAN
```

The sizes shown are the number of lines in the messages, not counting mail headers. For your convenience, this message has been specially formatted to make it easier to order the messages you are interested in. Just forward this message back to LISTSERV@PUCC (or LISTSERV@PUCC.PRINCETON.EDU) and fill in the line starting with "Print" (see below). Make sure to use the forward command of your mail program, not the normal reply function. There is a lot more the LISTSERV database functions can do for you - for instance, you can select all the messages with a particular subject in a single command rather than retyping all the index numbers. For more information, send an INFO DATABASE command to LISTSERV (you could add it before the line that says "Database search" the next time you order messages from the LISTSERV archive).

```
//    JOB
Database search DD=Orders
//Orders DD *
Select * in JAPAN.640-640
Print <type the numbers of the messages you want here>
/*
//    EOJ
```

Datafiles

The secondary activity of a list is file storage. If that option is activated, messages are stored in weekly or monthly logfiles. To review E-EUROPE messages for the month of January 1994, one would retrieve the file E-EUROPE LOG9401.

Besides logfiles, any other ASCII/text file can be stored for retrieval. As a side note, Listserv does not handle binary files such as WordPerfect documents, but this can be overcome by using UUENCODEing procedures. Check with your systems people about UUENCODE and UUDECODE.

These files can actually be peered-reviewed journals, working papers, UUENCODEd faxes or graphics, or whatever. The limit is set by the storage capacity of the host institution. The list editor can give file posting authority to others, but due to the technical nature of storing a file to a Listserv site, it is not open for public posting.

There is a command called AFD (automatic file distribution) which is infrequently used by most lists. It allows individuals to "subscribe" to a file, receiving copies of the file whenever the file is replaced or updated. This allows the creation of special interest groups within special interest groups.

For example, a JAPAN@PUCC subscriber started posting transcripts of the Radio Japan shortwave broadcasts. Since these were several hundred lines of text daily, several disinterested subscribers protested. The solution was to post the transcripts to a file called RADIO JPN. Those individuals who subscribe to this file using the AFD command receive copies of the daily replacement. Since the transcripts are not appended to the file, the distribution of the file by AFD represents a sub-specialty within the JAPAN group. The file is just a stored e-mail message.

Editor Beware

Editors should be aware of the vast differences of e-mail systems and costs among subscribers. A North American e-mail myopia is very common.

Routinely, American editors and subscribers assume that e-mail is a free good and treat it that way. They also assume that everyone on the list has the same quality of mail readers. Unfortunately, many overseas subscribers have to pay dearly for each message. Many mail systems do not provide full e-mail headers of FROM, TO or SUB-

JECT. The end result can be e-mail blurbs of "I agree" or "totally incorrect" without reference to the subject or who actually sent the message.

Individuals can also become very caustic behind their e-mail facade. E-mail flame wars are common on unmoderated public lists and will drive out serious discussion.

Large, unmoderated lists with high message levels will have relatively few COM (commercial-the subscriber must pay for e-mail) or international subscribers. CLINTON@MARIST is a good example. If the list is trying to attract COM and/or international subscribers, the messages will have to be relatively short and moderated by the list editor. One policy would be to limit e-mail message size, but post larger messages in the form of files for retrieval.

Setting Up A New List

With the large number of lists, is there one already in existence which would duplicate your proposed list? A first step would be to post an inquiry to NEW-LIST@NDSUVM1 (see the APPENDIX). If you find that there are several lists within the general topic area, query the editors of those lists, as well as post requests for information to the lists themselves.

The second step is to locate a host Listserv site. Review any directory of Listserv lists for host list nodes. Sending the command "lists" to the Listserv at that node will give you a directory of the lists at that node with a brief description of each, as well as the Listserv postmasters. Send your new list proposal to the desired Listserv postmaster requesting that university host your list.

Your proposal should have a detailed description of your list. This includes the nature of the list: Will it be moderated, e-mail only, public, private, etc.? Who are the potential subscribers? Does the editor (or editors) have the willingness to devote large blocks of time for the project? And of course, what will be the list's name?

When the list is accepted by a host institution, more mundane tasks must be performed. A one or two page description (promo) brochure should be written for distribution to potential subscribers. The mechanics of pc to Internet/Bitnet host communications must be mastered. The basic Listserv commands should be understood, with trial runs of adding and removing subscribers, forwarding e-mail, uploading and downloading files perfected before going public.

When you are ready to go public, post your document to NEW-LIST@NDSUVM1 and related lists and USENET newsgroups. Enjoy the activity which follows.

Appendix

NEW-LIST@NDSUVM1

The "NEW-LIST" list has been established as a central address to post announcements of new public mailing lists. In addition, "NEW-LIST" may be used as a FINAL verification before establishing a list (to check for existing lists on the same topic, etc.) or as a search address of last resort (only after doing your own searches).

Before posting an announcement or list search to the NEW-LIST list PLEASE send e-mail to LISTSERV@vm1.nodak.edu (or LISTSERV@NDSUVM1 on BITNET) with the command(s):

```
GET NEW-LIST FORMAT
GET LISTSOF LISTS
```

The first (NEW-LIST FORMAT) describes the format suggested for list announcements. The second (LISTSOF LISTS) gives some hints and pointers to more information on doing list searches.

If you have any questions please contact the list owner/editor:

Marty Hoag
hoag@vm1.nodak.edu or nu021172@NDSUVM1.BITNET
NDSU Information Technology Services
PO Box 5164 / IACC Room 206
Fargo, ND USA 58105
Phone: 701 237-8639 Fax: 701 237-8541

Strategic Planning to Implement a LAN from Ground Zero

Do Yeong Shin
Shaw University
118 E South street
Raleigh NC 27601
919-546-8352
dyshin@shawu.edu

Abstract

Small liberal arts colleges, like major research universities, need appropriate computing facilities, telecommunications, and networking for both administrative and academic areas. It may take years for a small liberal arts college to implement all of the above due to various constraints of the college. But the implementation planning could be done effectively in a short period of time before any implementation of hardware and software.

The officials of many small liberal arts college think that the Information Planning is so complicated that they cannot do it. But recently Shaw University, a small liberal arts college, with the help of a consultant from IBM, has successfully finished comprehensive studies of their information systems and services. The project took eight weeks. The procedures suggested by the consultant were somewhat complicated so that the persons selected to do the Information Planning Study had to spend time to comprehend the procedures. By eliminating the redundancy of the procedures suggested by the consultant and by revising the procedures, this study explored the more simplified procedures that would be as successful as the procedures Shaw's Information Planning Study Committee had followed.

The simplified procedures are designed to go through six phases. Each phase would be one week long. The possible activities of each phase are explored.

Introduction

Since the microcomputers have been introduced to the world in the 1970's, a variety of organizations including individuals have paid attention to the development of microcomputers. Among the organizations, there have been small liberal arts colleges whose primary missions are teaching rather than research. Their general environments are small compared with the major research universities so that these small liberal arts colleges cannot afford main frame computers. These financially healthy small colleges can afford minicomputers for the administrative data processing, but these computers are rarely available for classroom use. When the functions of stand-alone microcomputers have reasonably expanded with the moderate price tags, small colleges can start to acquire these machines for classroom use.

Background of the Problem

With an introduction of the Novell network, it has been common among the microcomputer users to classify the microcomputer ages into three phases: Boot Phase, Stand-alone Phase, and Network Phase. But there is no clear cut time distinguishing one phase from another. The transition from one phase to another depends solely on the institution's decision. The decision to implement the new phase is primarily based on the financial situation of the institution, competition with same size institutions, and finally the pressure from users.

Why a LAN is Never Installed

There are several reasons why a LAN has never been installed in a small liberal arts college. One reason is a lack of funds. Other reasons could be that the president of the college has never realized the importance of information processing and the effectiveness of teaching using a LAN, or the implementation of a LAN is stated in the college's long range plan as a low priority. But most college presidents perceive that the upgrading of stand- alone systems to a LAN is necessary for more efficient information processing and more effective teachings.

Unless the future of an institution is uncertain due to the overall financial situation, the financial situation may be a minor factor in implementing a LAN. The major factor is the recognition of the importance of a LAN by the president of a college. Also, stating the priority of implementing a LAN in the college's long range plan has to be adjusted to reflect the importance of a LAN.

Computer users cannot be excluded in this study. These users can be categorized in three groups: a group of administrative staffs, a group of students, and a group of faculty. A Computer Users Committee consists of representatives from the above three groups and from the professionals at the computing centers. Even though this committee is represented by appropriate persons, the passive activities of the committee have less impact on implementing a LAN. If the users committee is inactive, there is a lack of central control in the acquisitions of hardware and applications software. Thus, hardware and applications software are inconsistently purchased to meet specific demands for specific areas. When a system in a specific area does not work, the user cannot use another system in another area because of incompatibilities of applications software (and hardware).

Most small liberal arts colleges are not financially stable. Once these colleges recognize the importance of implementing a LAN for better information processing and effective teachings, they actively encourage faculty and staff to write proposals seeking external funding. In most cases, the proposals written by these well-qualified persons are rejected by the possible funding organizations because their proposals only concentrate on local views rather than the college's view as a whole.

Besides receiving funds from outside organizations, the perception of the president is that there are generally not enough internal human resources to know how to start a LAN plan. Convincing the president that there are human resources at the college is as difficult as receiving external funds. It is plausible to convince the president that the college has capable human resources as well as to emphasize the importance of implementing a LAN for better information processing, maintaining competition with same size of schools, and effectiveness of teaching. Sometimes it takes years to convince the president.

How to Start a LAN Plan

If the president, with all these failures, still cannot make the decision of implementing a LAN, another idea is to choose one of the vice presidents who might be able to influence the president. By convincing the vice president of the importance of implementing a LAN, (s)he can help the president make a decision about implementing a LAN. There are several strategies for this matter. One of them is a personal contact. Another option is to go with the vice president to the conferences or the seminars. By doing so, you can provide him with more information stressing the importance of implementing a LAN. This process can take years.

The next step is to convince the president that there are enough qualified human resources to do the initial investigation and to finish the project. If, either the president is not fully convinced, or there are not enough human resources to do the work, another option is seeking an external help. It is quite expensive for a small liberal arts college with a limited budget to hire a consultant. This problem can be solved by contacting the college or higher education program coordinator in a corporation, which usually provides free consultation with no obligation to their products.

Procedures and Activities

The first step toward the project is that the president has to establish a "Information Planning Study Committee". The possible members selected to the committee include all the functional Vice Presidents, Director of Academic Computing Center, Director of Data Processing Center (Administrative Computing Center), Director of the Learning Resources Center (the Library), and Faculty Representative(s).

The ultimate goal of the committee is to generate a detailed report which addresses the needs of the institution. To achieve this goal, the project will go through six phases.

Phase O. Pre-study Phase.

At this phase, the vice president who agrees with the necessity of a LAN explains the needs of Information Planning Study to the president who will sponsor the study and accept the results of the study. The vice president assures the president that (s)he will be periodically briefed throughout the study to ensure that the study's direction is consistent with his/her commitment to the college. The president determines the implementation period by consulting with the chief financial officer, and the college's short and long range plans. The president initiates the project by appointing Information Planning Study Committee members and the chair. The appointment letters are sent to the committee members. The chair schedules the first meeting and the committee members are notified.

Phase I. Questionnaire Analysis and Interview Schedule.

At this phase, two types of questionnaires - one for the General Survey and another for the Computer Centers Survey - are developed to find hardware and software resources at the computer centers, the users' interests in computers, experiences, and their prospects of computer usages, and so on.

Recommended Activities.

a) Determine the formats of two surveys; one for the General Survey, and another for the Computer Centers Survey.
b) Complete the General Survey and the Computer Center Survey forms ready for printing.
c) Distribute the surveys with the cover letters.
d) Committee identifies persons to be interviewed and makes groups of them.
e) Determine the possible questions for the interviewees.
f) Interview day schedule is determined.
g) Distribute the interview schedule along with the questions to be asked to each group member.
h) Questionnaires are collected.
i) Start compiling data for analysis.
j) Results of two surveys are ready.

Phase II. Interview, Problem Identification and Preliminary Report.

An interview is an exchange of information between the interviewer and the interviewee. It is a prime way to gather study facts and fill in the gaps resulted from the two surveys.

The problems identified in this phase are vital to view the current situation of the college and to determine the direction of the study. This important situation should be briefed to the president to ensure that the study direction is consistent with his/her commitment to the college.

Recommended Activities.

a) Discussion of the survey results.
b) Interviews.
c) Problems are identified.
d) The president is briefed.

Phase III. Recommendations and Implementation Planning.

The committee members specify the recommendations to satisfy both technical and non-technical requirements. They deliberate the constraints such as costs, personnel issues, political concerns, and so on. The benefits are identified. A detailed implementation schedule is developed.

Recommended Activities.

a) Recommendations are written.
b) Benefits are identified.
c) Implementation is scheduled.
d) Wrap-up session.

Phase IV. The final Edition of the Report.

The committee reviews the draft of the report and collects missing information. The editor is selected. If the editor is not a member of the committee, (s)he is invited to the final review session.

Recommended Activities.

a) The document is edited.
b) The remaining information is collected.
c) The edited copies are distributed to the committee members.

Phase V. The Final Report and Wrap-up.

At this last phase, the committee members, including the editor, meet together to determine the format of the final document. The format of the presentation to the president is determined and the chair arranges the schedule for the presentation.

Recommended Activities.

a) The Committee meets to finalize the report.
b) Print the final edited copy.
c) The Committee meets to finalize the format of the presentation to the president.
d) Present the study report to the president.
e) Revitalize the Users Committee.
f) Wrap-up and self-evaluation.

Conclusion

The Information Planning Study Committee has successfully finished comprehensive studies of Shaw's information systems and services in eight weeks. The author thinks that the report produced during the eight weeks could be generated with the same quality of work in six phases (weeks) in more simplified procedures.

The final copy of the report produced by the Information Planning Study Committee is the guideline for the college's overall information planning and detailed implementation schedule.

This document is also the resource for the persons in the college community who want to write proposals seeking for the funds for both hardware and software.

References

Burch, John G., Systems Analysis, Design, and Implementation, Boyd & Fraser, Boston, MA., 1992.

IBM Consultant's Notes

A Viable Alternative to Present Advising and Registration on the University of South Carolina at Spartanburg Campus

Stuart Shough
School of Business Administration and Economics
Chuck Morton
Student Services Coordinator
The University of South Carolina at Spartanburg
800 University Way
Spartanburg SC 29303
803-599-2571

In response to student discomfort with the present system of advising and registration, members of the faculty and staff of the School of Business Administration and Economics (SBA&E) have developed a "one stop shopping" approach that eases stress caused by this time-honored tradition. This system is based on the integration of four factors:

1) the student,
2) the adviser,
3) the University of South Carolina mainframe computing system, and
4) a "PC" interface with mainframe system.

Technical Aspects

The "PC" interface is made possible with a system of macros, and a database, developed for use with LOTUS 1,2,3. This interface allows the adviser electronic access to the most current records of student academic achievement. Utilizing 1,2,3 as a "database", student grades are downloaded from the university mainframe then displayed to reflect the student's academic progress. This gives adviser and student up-to-date information for advising purposes.

To meet the objectives of USCS, the system uses Personal Communications/3270 software and an IBM PC/3270 Communications Board. The PC/3270 board allows for communication to the university's ES/9000 where student records are stored and registration requests are logged. The operating system of choice is Microsoft Windows 3.1 loaded on an Intel 30486 microprocessor based system. This system can be easily adapted for other GUI (graphical user interface) operating systems such as Windows NT, IBM OS/2, and Apple Macintosh systems. Use with the GUI operating systems is best suited for this operation because it allows instant access to different applications (i.e., 1,2,3 and PC/3270) with just a "click" of the mouse.

The Academic Advising Process

When the student arrives for advisement he/she tells the adviser, who has activated 1,2,3 and mainframe communications software in Windows, his/her student identification number. This number, of course, is the same as the university uses to identify the student. The adviser enters the student number in a "dialog box" which in turn retrieves the student's academic record. It is at this point the adviser begins suggesting courses the student might want to consider. When the student and adviser agree upon which courses to take the adviser calls up a dialog box, which is tailored to accept course numbers (i.e. SBAD 225). Selected courses are entered into the "system". Once all courses are entered, a system of macros begins to search for all sections of courses entered and give the users a variety of formats in which courses can be displayed. Most students want to see a weekly layout of courses in a "schedule" type format. This system allows for such a view. Students can then choose a sequence of courses that best suit their personal schedules.

After students complete schedules the adviser "minimizes" 1,2,3 and "maximizes" the mainframe interface software. The adviser then enters courses, for the student, into the university's registration system. If section choices are closed, the registration system displays alternate section choices. A simple click of the mouse can minimize the mainframe interface and reopen the 1,2,3 system to rework the student's schedule. This design allows advisement and registration for courses in one place at one time. The student is happy not to have to run around looking for an adviser to "okay" another course selection, and the adviser is satisfied not to have to be annoyed with an endless barrage of frantic last minute requests to approve schedule changes due to closed sections of courses.

Students leave the advising session with a nicely printed copy of a course schedule and the knowledge that this schedule is logged with the university system guaranteeing the student a seat in the courses and sections selected.

Additional Enhancements

Other applications are being designed for this system including a prerequisite checking program, a course demand forecasting system, GPA analysis, a progress toward graduation analysis, and approval for "upper division" status. These applications will help in quicker decision making on student requests as well as making the administrative function of the School of Business "user friendly".

Conclusion

This system has proven to be very effective at getting student and adviser working together to design a well-rounded course schedule. In addition, screen design gives the appearance of working with actual student files thus creating the aura of user-friendliness. When students and advisers have used this approach to course scheduling the results have been very positive. Remarks, such as "I wish my adviser were using this" are not rare occurrences. Use of this "one stop shopping" approach has greatly affected student attitude toward advising and registration.

Where to Introduce C in the Small College Undergraduate Curriculum

Peter D. Smith
Mathematics Department
Saint Mary's College
219-284-4493
psmith@saintmarys.edu

Introduction

This paper is devoted to investigating the point in the curriculum when students should first be exposed to the C Language. We will discuss five different models for accomplishing this goal emphasiz ing the approach taken by Saint Mary's College. The five models are listed briefly as follows:

1. Introduce C first in an upper-division course such as Operating Systems, Programming Languages, or Algorithms. This model is the one adopted by most colleges that teach C at all. Some schools are moving from this to one of the other models.
2. Introduce C first in the Computer Organization (CS3) course. This is the approach taken by Saint Mary's, but it requires a complete inversion of the topics in the traditional Computer Organization course.
3. Introduce C or preferably C++ in the CS2 Data Structures course as this course begins to incorporate object-oriented programming (OOP) concepts to a greater extent. This is the most common model for colleges which attempt to incorporate the OOP paradigm into the curriculum more extensively than in just one upper division course.
4. Replace Pascal with Ansi C in CS1, but continue to teach the same procedural methodology as has become traditional in this course. Several textbook authors have taken this approach, but very few colleges seem to have made the switch from Pascal to C.
5. Teach OOP concepts from day 1 in CS1 using C++. This approach has more support pedagogically than the previous model.

Model 1: Introduction of C in an upper division course

Since C has become the industry standard for systems development work, it is an obvious choice of language to use in an Operating Systems course. The main problem with first introducing it in such a course is that there is very little time to teach C programming, given the complex theoretical concepts which must be covered.

Generally, students are given a brief overview of the syntax and structure of the language and then left to translate the structured design and programming style learned via Pascal to the development of programs in C. A bad program can be written in any language, but C's rich diversity of shorthand coding constructs makes it easier for students who learn only the syntax and structure of the language to write programs which are difficult if not impossible to maintain. This defeats the software engineering goals of the curriculum.

Some students, of course, do learn to program well in C, but if we wish to ensure that all upper division students maintain the good programming habits they learned in their lower division courses, we should consider moving the introduction of C earlier in the curriculum. [Roberts93, p.118]

Model 2: Introduction of C in the Computer Organization course

With proper planning, this can be an ideal place to introduce students to Ansi C. Ordinarily, the syllabus for this course treats digital logic, machine language, assembly language, and assemblers, linkers, interpreters, etc. in that order. There seems not to be a natural point in the traditional course at which to introduce C, but it would be an ideal language to use in the usual assembler course project.

Saint Mary's has found a way to introduce C in our Computer Organization course. The course was first offered in its present form in Spring 1992. It is taught on an alternating year schedule. Following a top down model [Scragg92, p. xxi], the course proceeds downwards starting with a review of the high level language model, and moving to the assembly language, machine language and digital logic models in that order.

The students who take this course for the most part have completed CS1 and CS2 and many have taken a course in Systems Analysis and Design or Data Base Organization. The purpose of the CS3 course is to take students through the levels of machine-human interaction. Treating the subject using a top-down approach makes the transition from other courses much more natural. The traditional bottom-up approach in which digital logic is treated first, followed by machine language and assembly language in that order forces the student to make a big jump from earlier courses.

We teach Assembly Language not so much because we think people will actually program in this language, but because it helps reveal the underlying hardware in a way not possible with high level language programming. Knowledge of hardware principles help computer science students better understand the discipline. It gives them another platform from which to view the field, and helps them become more thoughtful high level programmers as well.

We use Greg Scragg's Computer Organization: A Top-Down Approach as the text in our course and find it very effective. This text treats the high level model, the assembly level model, the machine level model, the digital logic model, and finally puts it all together in the integrated system model [Scragg92, p.xxi]. When reviewing the principles of high level languages, we introduce the students to Ansi C using the Borland

Turbo C++ development system. Since they are familiar with Pascal, we teach the parallels between the two languages, but also present their differences.

We are careful to emphasize good program design principles, so students become disciplined C program designers in a few weeks. During this time they are writing three homework programs including a project in which they implement an abstract data type. The course has a weekly closed laboratory where students work in pairs to solve problems by writing or extending C programs. This supervised lab experience becomes the central focus of the course and students realized its importance to their learning since they attended faithfully even though the lab was scheduled 2:30-4:30 on Friday afternoons.

After covering the essentials of Ansi C, we then teach assembly language (for the IBM PC 8088) where students embed assembly language statements into C programs. The Turbo C++ programming environment is especially helpful in this respect because students can open windows and watch the registers and memory locations change as they step through the program. We do not use assembly language I/O commands but do require all numbers to be converted by the programmer to character form before being written and after being read. Once we cover subprograms, we encourage students to call C program units in which the entire executable part is in assembly language. During this section of the course, they continue with the weekly labs, complete two more homework programs and begin working on an assembler for a simplified machine.

While the students are designing and coding their assemblers, we are covering the machine language model in class, including a careful study of the role and anatomy of an assembler. The course finishes with a brief look at digital logic, memory allocation, and linking and loading.

The students filled out a weekly lab evaluation form during Spring 93 and the results showed that they felt they had learned a lot from the experience without becoming overwhelmed by either C or assembly language. I am scheduled to teach the Computer Organization course again in Spring 95 and I intend to teach it again in a top down fashion. I am convinced that it is the way to go. I like the way all the parts of the course integrate well together. I attended Allen Tucker's post-conference SIGSCE workshop on courses 3 and 4 in the breadth-first curriculum this past March in Phoenix. He advocated a bottom up approach starting with digital logic and ending with an introduction to Lisp, but I still believe that Saint Mary's top-down approach makes more sense. [Tucker94, p.15]

The greatest difficulty in teaching the course was keeping in mind that Turbo C++ was also using the computer, resetting registers, adding function calls to the stack, etc. Often, the embedded assembly code would seem to be doing strange things as a result of the C++ function interference. There was not much documentation on Ansi C in the C++ manual, but I was able to get help over the Internet.

From my experience teaching Ansi C in this course, I would not want to move it any earlier in the curriculum. I was able to communi cate the importance of using good design in C programming to students who had two semesters of experience with structured programming and top down design in Pascal. Whether I could have helped them develop the same good habits if they were learning C as a first course is questionable.

Model 3: Introduction of C in Data Structures (CS2)

For many colleges, including Saint Mary's, CS2 is the place where object oriented programming (OOP) is first introduced. The abstract data types, traditionally studied in this course, can also be studied from an OOP point of view. The major problem for both the teacher and the student is the need to unlearn the procedural paradigm from CS1 and switch to the object-oriented paradigm.

Once a commitment to OOP is made in CS2, the problem of which language to use arises. There is a temptation to use the same language studied in CS1. In most cases this language is Pascal. In fact at a recent panel session on "New Models for the CS1 Course," at the March 1994 SIGSCE Symposium in Phoenix, nine-tenths of the attendees were still using Pascal in CS1, although about one-third were considering changing to C++. Even in the panel session "Using C++ in CS1/CS2," three-quarters of the attendees were still using Pascal. Saint Mary's currently uses Turbo Pascal 7.0 in both CS1 and CS2.

There are problems with most object oriented extensions to Pascal, however. It is hard to reap the benefits of reusable code, separation of declarations from definitions and private from public code where the declarations and executable code must be in the same program unit. It is interesting to note that the authors of a CS1 textbook, Stuart Hirshfield and Rick Decker from Hamilton College, pledged a firm commitment to Object Pascal at the SIGSCE Symposium in Indianapolis in February 1993 [Decker93, p.272] as they talked about converting their CS1 course to OOP, but at this year's Symposium [Decker94, p.54] they presented a paper giving reasons why C++ is the appropriate language for such a course. Their new textbook teaches OOP using C++ [Decker & Hirshfield94].

Model 4: Replacing Pascal with Ansi C in CS1

There are several reasons why Pascal is becoming an inadequate language for teaching modern programming concepts such as separate compilation, string manipulation, and use of functions as objects. Also, some of Pascal's constructs force its users to "program around the language" -- thus acquiring bad habits [Roberts93, p.117].

In spite of its deficiencies, most colleges are still using Pascal, although many are reevaluating this language choice as they decide whether or not to introduce the object-oriented paradigm in CS1. I will discuss this OOP decision more fully in the next model. Some colleges which retain the traditional procedure-oriented paradigm in CS1 have replaced Pascal with Ansi C, most notably Stanford University which began teaching C in all introductory programming courses in 1992-93 [Roberts93, p.117].

Some of the reasons that Roberts gives for changing to C include: a) the need to teach C at some point in the curriculum; b) the belief that it can be taught well in CS1 using software engineering strategies; and c) student demand to learn C in order to become more marketable. Stanford has developed a special I/O library 'simpio.h' which gives students access to the functions GETINTEGER and GETFLOAT. These functions read and return an integer and floating point value respectively, thus avoiding the need to learn SCANF. In a similar fashion, the Stanford faculty embedded a boolean type into a standard library to avoid the need to mention the equivalence of FALSE and 0. With such teaching strategies and with an emphasis on sound software engineering concepts, the switch to C was very successful in getting students to realize the value of abstraction. [Roberts93, p.120]

Roberts does not mention that any consideration was given to using C++ instead of Ansi C when the decision to replace Pascal in CS1 at Stanford was made. For a growing number of schools, the choice of language in CS1 is driven by changes in the programming paradigm taught. Some of these schools have chosen C++ because it supports object-oriented concepts better than Turbo Pascal 7.0. The next model will look at this decision process.

Model 5: Teach OOP Concepts from Day 1 in CS1 Using C++

This is by far the most radical of the five models. It involves not only a change in language but a complete new approach to programming. Paul Luker of California State University believes that the paradigm shift from structured programming, structured design and structured analysis to object-oriented programming, design and analysis is a revolutionary one.

He defines a paradigm as a methodology that creates models in the same way that scientific theories do. If the structured paradigm is no longer producing relevant models then a crisis has occurred and a new object-oriented paradigm is needed. He warns that everything should be done to produce better models from the old paradigm before jumping into a new one since revolutionary change brings with it significant costs in both software conversion and programmer reeducation [Luker94, p.57].

A number of colleges, especially small colleges, are facing a decision when and how to shift to an object-oriented paradigm. They cannot afford to jump on the bandwagon if it is not going anywhere. At the last SIGSCE conference, I overheard a textbook author comment that Pascal has survived the onslaught of PL/1 and Modula 2, so it can survive C++ and OOP. She chooses to ignore the fact that C++ already has a much larger industrial base than PL/1 or Modula 2 ever had. Students see the pages of want ads for C++ programmers and demand that we properly prepare them to enter the profession as they see it. Industry demand for Pascal, Modula 2 or PL/1 programmers was just not there.

Of course, our job is not to prepare students for entry-level programming jobs. If that were our role, we would still be teaching COBOL as our primary language. We shifted to languages that supported structured programming in the 70s because we believed that students must learn problem solving and design skills in a disciplined fashion. It is because object-oriented concepts provide another level of abstraction and a better platform for understanding the principles of the discipline of Computer Science that we consider introducing them into the curriculum.

Not everyone believes that OOP should be introduced in the first computer science course. Georgia Institute of Technology has decided to drop the teaching of OOP in their Introduction to Programming course because it was just too much for students to handle in a ten week course, and the language used, Turbo Pascal 7.0, "provided a convoluted implementation of object-oriented capabilities." [Shakelford94, p.9] Of course, Georgia Institute of Technology did not introduce OOP until later in the course and did not use a language like C++ which is more conducive to object-oriented programming.

Even some schools that have revised their CS1 course to emphasize the object-oriented paradigm, are not in favor of using C++. Luker at California State prefers Smalltalk, a purer object-oriented language [Luker94, p.60]. Meter and Miller at Carnegie Mellon University (CMU) are using Object Pascal (an extension to Pascal) to introduce object-oriented concepts. CMU had developed a Pascal Genie using Object Pascal five years ago to improve the user interface for their standard Pascal-oriented CS1 course, so it was easy to extend that Genie to handle the Object Pascal programming. The first two-thirds of the CS1 course at CMU is devoted to procedure-oriented concepts through arrays of records in standard Pascal and the last third of the course covers the basics of object-oriented programming, linked lists and sorting using objects instead of pointer structures. [Meter94, p.331]

Decker and Hirshfield at Hamilton College, on the other hand, are convinced that C++, because it is a hybrid language evolved from C, is more suitable for CS1 than a pure OOP language like Smalltalk. They found that Smalltalk was tightly coupled to its environment and to its powerful class libraries and had a steeper learning curve than languages which had procedural roots. Another reason for rejecting a pure OOP language was the fact that such a language does not support the explicit use of problem solving and top down design techniques as effectively as a hybrid language like C++. Learning these techniques is an essential and central component of CS1 [Decker93, p.272].

Decker and Hirshfield had first taught the object-oriented version of their CS1 course using Object Pascal, believing that Pascal, because it was a more familiar language, would require less learning overhead for both teachers and students, especially those who had a high school AP course [Decker93, p.273]. After one year they changed their CS1 language to C++. They found that Object Pascal did not allow students to fully appreciate the software engineering life cycle, especially the importance of reusability and separating declarations from definitions. This shortcoming turned out be an obstacle for students in learning good problem solving and top-down program design skills from an object-oriented point of view. It was not an insurmountable obstacle by any means, but C++ provided much more natural support for developing these skills. In fact, Decker and Hirshfield regard C++ as much more than a better C, but a better Pascal as well [Decker94, p.54].

The most powerful arguments for introducing the object-oriented paradigm to students from day 1 come from Decker and Hirshfield in my opinion. They believe that the object-oriented paradigm provides a conceptual framework on top of the development of algorithms which is a more natural support for human problem solving. "By describing classes one imposes an organization on a problem that not only clearly reflects those aspects of the real world being modeled, but also serves as a first crack at an algorithmic description of those same aspects [Decker94, p.53]."

When teaching students how to design a problem solution in a traditional CS1 course, we quickly reach a point where the algorithmic concepts of loops, decision structures and subprogram decomposition must be used. For many students this transition is very hard because it requires them to begin thinking like a computer. We give them lots of encouragement, telling them that it will soon all be clear, but this process has always been unsatisfy ing to me as a teacher.

For the last few years at Saint Mary's, we have asked students to begin the problem solving process by preparing a dialogue between a potential user of their program and the program itself. This process has seemed quite natural and mirrors the student's own questioning process as she strives to understand the problem. During this dialogue certain data objects arise naturally and are identified and investigated. Then students begin the traditional top-down algorithm development process. It is at this point in the problem solving process that the object-oriented paradigm could be of most value. By describing classes for the data objects, and then associating functions with classes, students can more naturally break the problem down into more manageable subproblems. In the end, of course, algorithms for the functions must be developed, but pushing their development off to the end of the refinement process should make the transition easier for students.

Decker and Hirshfield emphasize the importance of making CS1 a laboratory centered course. This is true to our experience at Saint Mary's as well, although we have not yet introduced OOP into our CS1 course. Building the course around the closed lab experience, students can first "use the object world by interacting with sample programs that are provided to illustrate specific OOP features. As they become more familiar with objects, students are exposed to the details of the world so they can read the implemen tations of the objects they are using. Then, they are shown how they can modify and extend the existing object world to suit the needs of a particular program. Finally, they are taught to define their own objects [Decker93 p.272]."

There are a number of other reasons for starting students off right away in the object-oriented paradigm. There is no need for a paradigm shift later in the curriculum when OOP is finally introduced. Experience has shown that students develop better programming habits, and can tackle and solve more sophisticated problems using OOP. When I first decided to write this paper, I was very skeptical about exposing beginning students to the complexities of OOP. But after considering the arguments both pro and con I am convinced that this is the way to go. We can keep all that is valuable in the traditional development of problem solving skills, while using OOP to provide a more natural interface to algorithm design.

Conclusions

In summary, this paper has provided five models for introducing C (or C++) in the undergraduate curriculum. I believe that introducing Ansi C in the first course is a bad idea. Pascal should be retained until the decision is made to evolve the course out of a procedural paradigm.

The reasons for introducing the object-oriented paradigm using C++ right away in CS1 seem to me to be compelling. Whereas the introduction of C++ and OOP in CS2 is probably a necessary interim solution, I believe that both CS1 and CS2 should be revised to allow object-oriented concepts to pervade. This is a major undertaking, especially for small colleges with limited staff. Fortunately, good textbooks supporting OOP early in the curriculum are beginning to appear.

I am also convinced that the best way to teach the Computer Organization course is in a top-down manner using Ansi C as the high level language component. As mentioned above, I do not recommend introducing C any earlier than CS3.

In writing this paper, I hope to promote discussion at the ASCUE Conference and thereafter. I have had very little experience teaching OOP concepts and am looking forward to the response of my more experienced colleagues.

References

[Decker93] Rick Decker and Stuart Hirshfield, "Top-Down Teaching: Object-Oriented Programming in CS1," Proceedings of the 24th SIGSCE Technical Symposium on Computer Science Education, March 1993.

[Decker94] Rick Decker and Stuart Hirshfield, "The Top Ten Reasons Why Object-Oriented Programming Can't be Taught in CS1," Proceedings of the 25th SIGSCE Technical Symposium on Computer Science Education, March 1994.

[Decker and Hirshfield94] Rick Decker and Stuart Hirshfield, The Object Concept: An Introduction to Computer Programming Using C++, PWS Publishing, Inc., Boston 1994.

[Luker94] Paul A. Luker, "There's More to OOP than Syntax!," Proceedings of the 25th SIGSCE Technical Symposium on Computer Science Education, March 1994.

[Meter94] Glenn Meter and Philip Miller, "Engaging Students and Teaching Modern Concepts: Literate, Situated, Object-Oriented Programming," Proceedings of the 25th SIGSCE Technical Symposium on Computer Science Education, March 1994.

[Roberts93] Eric S. Roberts, "Using C in CS1: Evaluating the Stanford Experience," Proceedings of the 24th SIGSCE Technical Symposium on Computer Science Education, March 1993.

[Scragg92] Greg W. Scragg, Computer Organization: A Top-Down Approach, McGraw Hill, Inc., New York 1992.

[Shakelford94] Russell L. Shakelford and Richard LeBlanc, "Integrating 'Depth First' and 'Breadth First' Models of Computing Curricula," Proceedings of the 25th SIGSCE Technical Symposium on Computer Science Education, March 1994.

[Tucker94] Allen B. Tucker and Peter Wegner, "New Directions in the Introductory Computer Science Curriculum," Proceedings of the 25th SIGSCE Technical Symposium on Computer Science Education, March 1994.

Using Maple in the Calculus Sequence

Thomas Smith, Charles Stolze, Patrice Tiffany, Kathryn Weld
Department of Mathematics and Computer Science
Manhattan College
Manhattan College Parkway
Riverdale NY 10471
718-920-0253
tsmith@manvax.cc.mancol.edu

Introduction

During the past decade the mathematical community has engaged in an intense curriculum reexamination, primarily concentrated on the teaching of calculus. Spurred by calls from the Mathematical Association of America and by grants from the National Science Foundation, various groups have studied ways to change the teaching of a subject whose curriculum and methodology have been largely in place for over a hundred years. The goals of these efforts have been to make calculus more lively, to introduce more realistic problems, to involve students more actively in the learning process, and to take advantage of the technology incorporated in graphing calculators and computers.

This paper will focus on the efforts of faculty members from Manhattan College to bring this new technology into the calculus sequence. After experimenting with computer algebra systems such as Derive and Mathematica, the decision was made to use Maple which is a software package developed by faculty at the University of Waterloo and designed to be run on microcomputer systems. This choice was based on several factors - powerful symbolic algebra, graphing and numerical capabilities, ease of use, relatively inexpensive cost, and a networked version which performed quite well. The college has Maple running on a Novell network of IBM Value Point 486 machines. The network services two microcomputer classrooms (23 machines in each), a computer lab (for homework), as well as machines in each of the faculty offices.

A grant from the General Electric Foundation allowed the Department of Mathematics and Computer Science to offer faculty a one week workshop during the summer of 1993. At the workshop faculty were instructed in the use of Maple and got to see the software used to solve typical problems which arise in calculus. Two of the authors (C. Stolze and K. Weld) were the principal instructors, each having used the system informally in classes during the previous academic year. One of the outcomes of the project was the decision to use Maple in a small number of pilot sections of calculus I and II during the 1993-94 academic year.

Pilot Project

The pilot project consisted of three sections of first year calculus (taught by T. Smith, C. Stolze, and P. Tiffany) and one section of Honors Calculus (taught by K. Weld).

Each of the four sections met in a regular classroom for approximately one-half of the class time and in a computer classroom for the remaining time. The four instructors used different approaches to the lab time with each designing individual labs and group projects. Although the faculty members developed their own materials, the group met frequently to discuss the different approaches and to share successes (and failures). There was considerable variety in terms of the number of projects assigned, the length of the projects, and the method of student participation. Some of the labs involved students working in teams (similar to the way in which students in physics and chemistry labs are assigned lab partners), others had students working individually. One instructor had the students work in pairs but required that the pairs be rotated each week to insure that students got to meet and work with a large collection of their peers. To provide additional data for later evaluation, two of the faculty used the Windows version of Maple while two used the DOS version. As can be seen, faculty were very much free to experiment with any approach with which they felt comfortable.

Critical Issues

The use of a computer algebra system for the teaching of calculus raises several critical issues.

1. How should the curriculum of the course be modified? What topics should be eliminated, or at least down played? How do you cover all of the necessary material while sacrificing traditional instructional time in favor of lab time?
2. How do you ensure that students will master the same algebraic and problem solving skills traditionally emphasized in the standard course? This is a particular concern of faculty who will be teaching these students in later courses which have calculus as a prerequisite. (At Manhattan College this concern has been expressed most strongly by faculty in the Engineering School since the majority of the students in the calculus sequence are from that school.)
3. How do you avoid the black box effect? How do students learn to use the computer to improve their understanding of the material as opposed to using the computer as a crutch? How do you incorporate the technology so that it becomes an enhancement rather than a replacement?
4. How, in fact, do students learn mathematics? The traditional syllabus emphasizes algebraic techniques. Use of a computer algebra system allows equal weight to be given to algebraic (symbolical), numerical and graphical techniques. Does this facilitate an understanding of the ideas? Or do students need to master one form of knowledge of the material (algebraic, for example) before they have the maturity to interpret concepts graphically or numerically?

Results

Based on our experience over the past year we have begun to formulate responses to the issues outlined above.

1. The curriculum of the standard calculus course will be modified but it will take time to arrive at a revision which achieves a broad consensus. A computer algebra system does allow the instructor to reduce the time spent on several topics - most notably curve sketching, numerical integration and several of the techniques of integration. The graphical and numerical capabilities allow students to explore topics like limits and continuity in more detail and for more difficult functions than could be done previously by hand. The syllabus can be covered but there must be a judicious selection of material. Some of that material is better done in the old way and some is much better done using the computer algebra system.
2. One approach to the maintenance of traditional skills is to have students do some of the projects by hand and then to check the results using the computer. On top of that, the examinations can be traditional in nature, requiring that students demonstrate the usual skills.
3. To avoid the black box effect, it is necessary to include in the lab projects questions which require that the students draw conclusions from what they are doing. It is not enough to ask them to produce answers. The meaning of the answers must be emphasized. A somewhat more drastic technique is to assign a small number of problems for which the computer algebra system fails, either because the algorithms built into the system are not up to the task or because there are bugs in the code. (Alas, there are bugs in the code.)
4. The answer to the question of how students learn mathematics seems to vary from student to student. It is clear, however, that care must be given to lab design. In attempting to avoid the black box syndrome it is easy to make labs too hard. Many of the calculus reform texts currently on the market replace "techniques of computation" (the dreaded cookbook approach) with "lively" topics. An intuitive expository style is sometimes justified under the name "graphical" thinking. There is much of value in this approach, but it must be remembered that for many students the concepts are not intuitively easy. Translating intuition into computation is what makes mathematics hard. Computation does allow students a feeling of mastery and confidence. While we feel Maple is a valuable tool, and that the graphical and numerical capabilities of the program greatly aid in teaching many topics, we have become more traditional in our approach after the first year's experience.

Beyond the critical issues, several results have surfaced.

- the students have had no trouble learning to use Maple. Although Maple does contain its own programming language, there has been no attempt to teach the students to program. Each lab assignment may involve a few Maple commands and these are given to the students via an example which they do prior to beginning the actual problems which they will submit. (Some sample labs are included in the Appendix.) Students are not asked to memorize any of the Maple commands for examinations.

- the graphical capabilities have helped students to "see" connections more clearly than before. For example, the significance of the signs of the first and second derivatives in curve sketching is brought home forcefully by having students plot the function and its first and second derivatives on the same graph (an extremely simple process in Maple). Similarly, the approximation of areas under curves by the areas of rectangles is illustrated both graphically and numerically in a small fraction of the time usually needed for this idea. It is easy to "see" that the approximation improves as the number of rectangles increases.

- the numerical capabilities allow students to construct tables which illustrate limits of functions and limits of difference quotients. The notion of a function approaching a fixed number as we approach a fixed point is easier to demonstrate, particularly for nontrivial examples.

- the use of a computer laboratory as a teaching instrument allows an instructor considerable flexibility. The four faculty members involved have found that their individual teaching styles have been maintained. While some of us welcome the chance to use a laboratory to create a less structured and more student centered teaching environment, others have used the laboratory in a very structured way. This has encouraged us to feel that the use of Maple in teaching calculus could be embraced by all mathematics faculty.

Assessment

An assessment process, designed and monitored by a faculty member not teaching any of the pilot sections, has been put into place. The process includes surveys of student attitudes toward the computer oriented approach to calculus and an attempt to determine how well students who have taken the pilot sections do in later mathematics courses. The latter part of the process will continue over several years.

Conclusions

The four faculty teaching the pilot sections agree that the new approach has great potential although there are still many problems to be resolved. Curriculum changes will evolve as faculty gain more experience dealing with this type of course.

The positive response of the four faculty has led the department to decide to teach all of the sections of elementary calculus next semester using Maple. As preparation for this expanded thrust, the department will offer a short workshop this summer in which faculty who have not taught using Maple will be able to experiment with the techniques developed this year. The lab assignments used by the four faculty will be gathered into a loose leaf binder to serve as a resource for those entering the project.

Faculty new to the project will be encouraged to try their own ideas but they will have the advantage of avoiding the pitfalls experienced by those who taught the courses this year.

Appendix - Sample Labs

MATH 103 - 08 LAB #3 - September 21, 1993 Dr.T.Smith

For this lab, you must be familiar with the material in sections 1.11, 1.12, 2.1, 2.2, 2.3 in the text.

1. Rational Functions and Continuity

For each of the following functions use appropriate plots to answer the questions.
a) What happens as x -> -2 from the left? from the right?
b) What happens as x -> 2 from the left? from the right?
c) What happens as x -> - infinity? + infinity?
d) What are the vertical asymptotes?
e) What are the x-intercepts?
f) The graph is symmetric with respect to what?
g) Where is the function discontinuous?

A) `f(x) = x/(x^2 - 4);` B) `f(x) = (x^2 - 1)/(x^2 - 4)`

For problems 2, 3 and 4 you will need three procedures which are available on the file server. Follow these directions to copy the procedures from the file server to your floppy disk. Later you will read the procedures into Maple from your floppy.

At the DOS prompt, C:\TEMP>, insert your floppy into the disk drive, type a:, and hit the ENTER key. At the A: prompt type

`COPY H:\HOME\MATH\TOMSMITH\LIMTABLE` and hit ENTER.

Leave a space after the word COPY but do not leave spaces anywhere else. This command will copy the file named LIMTABLE from the file server to your floppy. Do this again for the two other procedures.

`COPY H:\HOME\MATH\TOMSMITH\SECLINE` and hit ENTER, and

`COPY H:\HOME\MATH\TOMSMITH\TLINE`and hit ENTER.

Check that the files are on your floppy by typing dir at the a: prompt.

Switch back to the C:\TEMP> prompt by typing C:

When you are in Maple, you may read in the appropriate file by typing the command

``> read (`a:limtable`);`` - note here the use of the backslash located above the Tab key. This read command will copy the procedure from your floppy into the Maple environment.

2. Limits - In this exercise we wish to check whether a certain function has a limit at a given point by examining values of the function near the point. (The function need not be defined at the point.) To help us we will use a table of function values. The table will contain values f(a+h), f(a-h) for several small values of h. As h -> 0, a+h and a-h approach a so we are examining values of f near a (on both sides).

For example, if f(x) = x^2, and a = 2, we expect that the values of f near a = 2 should be close to 4 (i.e. lim x^2 = 4 as x -> 2). To verify this, define f(x) = x^2 and call the procedure

`> limtable(f,2);` {Note the function values as h gets small.}

Use the procedure limtable to determine if the function below has a limit at the given point. If so, what is the limit? If not, why not?
a) f(x) = sin(x)/x; a = 0;
b) f(x) = (1 - cos(x))/x, a = 0;
c) f(x) = (x^3 - x^2 - 4*x +4)/(x - 1), a = 1;
d) f(x) = x/(x^2 - 1), a = 1;
e) f(x) = (x^(1/2) - 2)/(x - 4); a = 4;
f) f(x) = (exp(x) - 1)/x, a = 0.

Note: you may wish to check your answers by plotting each function near the indicated point.

WARNING: Be careful when using computer generated material. In some cases, the computer algorithms do not work. Try
f(x) = (2^x - 1)/x at a = 0
and compare the table with example #7 on page 134.

3. Secant Lines - For this exercise you will need the procedure secline on your floppy.

We use secant lines as approximations to the tangent line to a curve at a point. For a given function and a given point, the procedure secline will graph the function and four secant lines near the point. Each secant line has a slope of (f(a+h) - f(a))/h, where h = 1, .75, .5, .25. As h approaches 0 the secant line should approach the tangent line (if the latter exists). Refer to the discussion on page 131 of the text. As an example, define f(x) = x^2 and call the procedure secline by typing

> secline(f,0), to see the plot of f and the four secant lines near a = 0. See if you can determine which secant line corresponds to h = 1, h = .75, etc.

For each function below, answer these questions. As h -> 0, are the slopes of the secant lines increasing or decreasing? Can you guess what the slope of the tangent line will be? Print out the graph for each and label the secant lines (h = 1, etc.).
a) f(x) = 4 - x^2; a = 0;
b) f(x) = 4 - x^2; a = -1;
c) f(x) = x^3, a = 0;
d) f(x) = x^3, a = -1;
e) f(x) = cos(x), a = 0.

4. Tangent Lines - This exercise is similar to exercise 3 above, except that the procedure tline graphs a function and its tangent line at a given point. For each function below is the slope of the tangent line positive or negative? Is the curve increasing or decreasing at the point?
a) f(x) = sin(x), a = 0;
b) f(x) = sin(x), a = Pi;
c) f(x) = 4 - x^2, a = 0;
d) f(x) = 4 - x^2, a = -.5.

What is the moral of this story?

DUE: Tuesday, September 28, 1993 at the beginning of class.

MATH 103 - 08 LAB #4 - September 28, 1993 Dr. T. Smith

1. This exercise is a continuation of Exercise #3 in Lab #3. In the previous exercise you plotted the graphs of several secant lines and attempted to determine the slope of the tangent line. In this exercise, you will use a table of difference quotients for the same purpose. The table dqtable is saved as a procedure on the system file server (consult the notes from Lab #3 on how to copy this to your floppy and then read it into Maple). You will call the table using the command > dqtable(f,a).

For each of the functions in exercise #3 of Lab #3 use the table to determine the slope of the tangent line to the function f at the point a. Write down the equation of the tangent line to the graph of f(x) at x = a. (No plots necessary.)

2. Maple allows us to simplify the work involved in calculating limits of difference quotients. We can use this to calculate the symbolic derivative of a function (i.e. the derivative at any point). For each of the functions below, use the following sequence of commands to produce the function f'(x).

```
> f(x+h) - f(x);
> "/h;
> simplify (");
> limit (", h = 0);
```

Note that the last command computes the limit of the difference quotient at h = 0.

a) f(x) = x^2;
b) f(x) = x^3;
c) f(x) = 3*x^4 - 4*x^3 + 2*x + 1;
d) f(x) = 2^x;
e) f(x) = exp(x);
f) f(x) = sin(x);

3. Maple further simplifies our work by combining the commands in exercise #2 above into one command. For a given function f, the command D(f) will produce the symbolic derivative (i.e. the derivative of f(x) at any point). For the functions below, plot f and D(f) on appropriate axes to answer these questions:
a) On what interval, or intervals, is f(x) increasing? decreasing?
b) On what intervals is f'(x) > 0? < 0?
c) What is the moral of the story?

Note: Hand in plots for each part.

i) f(x) = x^2 - 8*x + 9; ii) f(x) = x^3 - 3*x + 1;
iii) f(x) = sin(x) on [-Pi..Pi].

4. Do exercise #28 on page 148.

DUE: Tuesday, October 5, 1993 at the beginning of class.

MATH 103 -08 LAB #9 - November 16 ,1993 DR.T.SMITH

In this lab we will use Maple to assist us in determining the area bounded above by the graph of a function and below by the x-axis. As a first approximation, we will estimate the area by using a certain number of rectangles. We improve the estimate by increasing the number of rectangles. Finally, we will take the limit as the number of rectangles approaches infinity.

As an illustration, begin by defining

```
> with(student); {This accesses the package of routines we need.}
> f := x -> x^2;
> leftbox (f, x = 0..1, 10); {This will show us what an estimate looks like using the left hand endpoint and 10 rectangles.}
> leftsum (f(x), x = 0..1, 10); {This is the symbolic form of the sum of the areas of the 10 rectangles.}
> value ("); {This is the value of the approximation using the 10 rectangles. Record this value.}
> rightbox (f, x = 0..1, 10); {This will show us the estimate using the right hand endpoint in each of the 10 rectangles.}
> rightsum (f(x), x = 0..1, 10);
> value ("); {Record this value. Since the rectangles using the left hand endpoints are all below the curve and the rectangles using the right hand endpoints are above the curve, the correct answer must be between the two numbers found. We improve the approximation by increasing the number of rectangles used. Repeat the leftsum and value commands for n = 100 rectangles and record these answers.}
> leftsum (f(x), x = 0..1,n); {This produces the sum of the areas for an arbitrary number of rectangles.}
> limit (", n = infinity);
> value ("); {This should be the exact answer. To check it, do
> rightsum (f(x), x = 0..1,n);
> limit (", n = infinity);
> value ("); {This should agree with the answer found above.}
```

Repeat what we just did for

1. f(x) = x^3 on [0,2]
2. g(x) = cos(x) on [0, Pi/2]
4. h(x) = exp(x) on [0, 1]
2. F(x) = sin(x) on [0, Pi]. Note here that the left rectangles and the right rectangles are not always on one side of the curve. Why is this? How is this function different?

In each of the four problems, hand in a plot for the leftbox and rightbox using 10 rectangles and a record of the answers for each calculation (i.e. 10 leftsum approximations, 10 rightsums, 100 left and right sums, and the two limits. These last two numbers should agree for each function and should give us the exact area under the curve.

Due: Tuesday, November 23th, 1993 at the beginning of class.

Centralized Computer Direction Without Centralized Computer Control

Robin M. Snyder
Director of Academic Computing
Associate Professor of Information Systems
The Byrd School of Business
Shenandoah University
1460 University Drive
Winchester VA 2601
703-665-4938

Abstract

In the old days, computers were expensive and people were inexpensive. At that time, the Director of Computing controlled a centralized resource and people wanting computer support were at the mercy of the Director of Computing, who wielded considerable power in this area. Due to their cost, information systems and computer technology were available to a selected few. Today, computers are inexpensive and people are expensive. People wanting computer support can go out and buy the computer hardware and software themselves, but, in the face of a fast-changing and sophisticated technology, have difficulty in achieving their goals within budgetary constraints (due to ill-advised decisions and general lack of expertise in the area of information systems and computer technology). This paper discusses the_problem of providing centralized information systems and computer direction and leadership without requiring centralized information systems and computer control.

Introduction

Before we can solve a problem, we must first (correctly) identify the problem. In the area of information systems and computer technology, the typical problem that faces many institutions can be explained as follows, using the Mythical Typical University (MTU) as the institution (go ahead, insert your own institution's name and see how it fits).

Problem statement: There is no one person at MTU who is responsible for information systems and computer technology, and, if there is, this person does not have authority (resources) to accomplish much. This creates problems in arriving at a consensus for decisions that are global in impact, in creating systems that transcend departmental boundaries, and in achieving cost savings through coordinated efforts.

If you relate to this problem statement, then you realize that the problem is a sensitive one and needs to be handled carefully. So remember, MTU is a combination of many institutions, academic, business, government, and others. In particular, do not feel that it is an indictment of any particular institution. The goal here is to identify problems and propose potential solutions.

Now, information systems are more than just a computer. Information systems are a combination of hardware, software, information, policies, and people that combine to solve a problem of interest, usually by adding value to the organization in which the information system is established. Today, computer technology drives information systems to the extent that the two areas are_difficult to separate. For this reason, the terms "Information systems and computer technology" will be used together. And this combination of hardware, software, information, policies, and people that we call information systems and computer technology do not spontaneously appear and continue to work in an orderly manner without some driving force. So, our beginning assumption is,

#1: Someone needs to be in charge of information systems and computer technology.

We do not need centralized control of information systems and computer technology at MTU, but we need centralized direction, or leadership, of information systems and computer technology. That is, we do not need someone to tell everyone else how to do something (which never seems to work well), but we need someone who can take a proactive stance in bringing proper focus_to pertinent problems such that a workable group consensus can be achieved, and enforced once agreed to. Such direction and consensus provides a foundation for future decisions (and future consensus).

We will call this person the Chief Information Officer (CIO).

> ...the VP of MIS (sometimes called the chief information officer, or CIO)reports directly to the CEO. This arrangement reflects the importance of information and information systems to business organizations today. In prior years, the MIS department reported to the VP of Finance. This structure proved to be unworkable, primarily because organizations need information systems beyond those of concern to Finance. (Kroenke, p. 608)

We cannot just pick anyone and expect that person to fit in. Information systems and computer technology are based on the principles of computation (also called computer programming). Why? Well, the analysis of the effectiveness and efficiencies of information flow within an organizational structure is, mathematically and scientifically, the same, whether the solution to a problem of interest is to be implemented with or without computer hardware and/or software. And,

> ... the leader of a programming group will, in most cases, emerge as the individual who is most technically competent at each stage of the project. If this is not recognized by higher management and an unwanted leader is_imposed on the group, this is likely to introduce tensions into the group._The members will certainly not respect the leader and may reject group_loyalty in favor of individual goals. This is a particular problem in a fast-changing field such as software engineering ... (Sommerville, 1989, p. 39).

And when we are creating an organizational information system, we are using the principles of software engineering. So the CIO should be an information systems and computer technology person, but the CIO is more than just a computer person. In fact, today's (information)

> system developers are as likely to recommend a change in organizational structure as they are likely to recommend a particular type of computer hardware (Kroenke, p. 91).

And changes in organizational structure require support from the very top of the organization. So the next time you request assistance from a capable_computer and information systems person, you may get more feedback than just an answer as to "What computer hardware and software should I buy?". Information systems and computer technology are more than just computer hardware and software. This means that,

#2: Organizational change will be required. Are you ready and willing?

Now, let us suppose that MTU desires to consolidate information systems and computer technology. We could compose a job description consisting of a laundry list of duties and responsibilities, pick a job title (Director of Computing, Chief Information Officer (CIO), etc.), and what not. Something like the following.

> Wanted: Chief Information Officer (CIO).
>
> Must be faster than a speeding bullet, more powerful than a locomotive, and able to leap tall buildings in a single bound. Must also have an encyclopedic knowledge of every aspect of LAN and communications technology, the ability to diagnose problems with a minimum of equipment, and be capable of learning and supporting every new product without training. The successful applicant will not require any staff, holidays, or much sleep, and should not plan to have any kind of personal life. The job pay is OK, but don't expect too much recognition or understanding of what you actually do from anyone in the organization other than the other people in MIS. Ability to work miracles preferred. (Gibbs, 1993, p. 26).

When I arrived at my present position as Director of Academic Computing and Associate Professor of Computer Science, with some high-powered qualifications, the VP for Finance asked me, "What will happen if you do what you are saying you will do and then leave. Can we replace you?." After a few days, the following revelation hit me. What the VP for Finance is looking for is stability, and that means management. The least common denominator type of management, so that when the CIO leaves, someone can be easily found to replace the CIO. And this least common denominator can be called mediocrity. That is, do what everyone else is doing so that anyone can change jobs. Well, the question to be asked is, "Is what everyone else is doing working?". If you publicly ask the people in charge, the answer is almost certainly "Yes, definitely.". If you ask them privately, or ask the person in the street, the answer is closer to, "No, not really.".

My reply to the question of management is this. The fast changing pace of information systems and computer technology dictates that proactive leadership and management, not just management, is needed. To use a sports analogy, a successful head football (or basketball, etc.) coach can leave behind all the assistants, the playbooks, the films, the players, everything except himself. And the team can completely fall apart (or slowly disintegrate). Any new head coach will most probably evaluate the situation and begin molding the team to his own particular vision of what will lead to success. And that takes time. That is, in large part, the state of the art in information systems and computer technology. Yes, there are rules of the game, established by decades of business, scientific, and engineering work done in the field of information systems and computer technology. And there are rules that cannot be violated without paying a penalty. Rules such as the noncomputer-checked redundancy problem (Snyder, 1994). But there remains substantial leeway in how such rules are applied. Just as a head coach must follow rules, both written rules in the rulebook and unwritten rules or principles (for example, physical constraints of people). But the head coach still has substantial leeway in applying those rules and principles in practice.

The situation in the old days (where computers were expensive and people were inexpensive) lent itself to management. Good management cannot guarantee success but bad management almost always guarantees failure. This should be modified in the face of today's fast-changing and far-reaching situation (where computers are inexpensive and people are expensive). Good management is not enough. Proactive leadership is required. Good leadership cannot guarantee success, but bad leadership almost always guarantees failure.

#3: Leadership and management, not just management, are required.

The recommendation: Find a CIO that has the capability to provide proactive leadership, who you have confidence in, and who you can trust. Trust with resources like money! Give that CIO responsibility and authority, but not without accountability. And wait. It will take years. A long term contract is in order (just as many coaches are given long term contracts). If you make the wrong decision, the institution will suffer. But if you make no decision, you will certainly be at a competitive disadvantage, since the field will not wait given the fast changing pace of information systems and computer technology). And remember, many successful head coaches are very conservative (but know when to take risks).

Responsibility

What are some responsibilities of the CIO? Probably something like the following.

- To maintain a global perspective of all aspects of information systems and computer technology in terms of the real and perceived goals of MTU. This includes creating/maintaining a system to keep track of both resources and goals.
- To serve as an advisor and/or troubleshooter for complicated and sophisticated problems/systems involving information systems and computer technology. As an advisor to avoid ill-advised decisions (that could be very expensive) and as a troubleshooter to make systems work (few systems work out-of-the-box).
- To become involved proactively in problem identification, definition, design, implementation, and evaluation of all problems/systems that involve information systems and/or computer technology. Such systems often require organizational change and this needs coordination, consensus, and support at the highest levels

of the organization.

- To determine a yearly information systems and computer technology budget (Snyder, 1993a) and to monitor that budget (in conjunction with the VP for Finance).
- To maintaining an up-to-date written record of global and local decisions (based on consensus). This is important since any new employee whose job will be impacted by these decisions (based on a consensus of which they were not a part) may need to know about these decisions in order to make an employment decision. The CIO is the only person in a position to maintain such a record.
- To keeping those at MTU informed as to what is happening and to serve as a point of contact to outside institutions as needed.
- To serve as an educational resource in the area of information systems and computer technology, as needed. Must be able to communicate complex and sophisticated ideas and concepts in understandable terminology.
- To perform other duties, as needed (the proverbial catchall phrase).

Some implications of these responsibilities can be summarized as follows.

- In order to keep up with information systems and computer technology, to keep a perspective on our true business (educating students), and to develop systems on a demand-driven basis (to develop a working system, one must get intimately involved with the problem to be solved), the CIO should be involved with the business of the institution. In the case of MTU, this is educating students. Besides, we would expect someone intimately involved with information systems and computer technology to benefit from the teaching of this subject (who, having taught a course, has not learned more about the subject area?).
- The CIO should write, publish, and present papers at relevant conferences and meetings (the ability to communicate is very important). These papers should relate directly to the information systems and computer technology being used at the institution. This is important from a professional perspective, providing feedback on current work from peer institutions across the country, allowing the CIO to keep up with the fast-paced field of information systems and computer technology. Support should be provided for such endeavors.
- The CIO should identify, define, design, implement, and evaluate real systems that solve real problems. This is a challenging and rewarding activity. Having done this (whatever the problem), it is but a mere formality for the CIO to write a paper describing the system and have it published. Other people will want to hear how such problems have been addressed and the CIO will receive feedback (and recognition) for the work performed. Note: People for whom systems are built tend to feel less threatened when they realize that their jobs will not be replaced; the new system will make their life easier (less work and/or more productivity), and that they will be responsible for managing the system after it is complete. Such systems should facilitate global concerns of which the people involved may not even be aware.

Note that some academic institutions may include grant writing as a responsibility.

Authority

OK, you say. What next? Well, if the leadership is to be effective,

#4: The authority must match the responsibility.

By authority, we mean the resources (time, money, people, etc.) necessary to accomplish the responsibilities. A common shortcoming is to give responsibility without authority (no time, money, people, etc.) - a no-win situation for both sides. The authority needed in order to achieve these responsibilities include the following.

- The CIO should officially report directly to the CEO, or president, and have the full support of the CEO. Unofficially, the CIO works with anyone and everyone who has information systems and computer technology needs. In particular, the CIO does not directly have the power to tell anyone to do anything. However, in practice, the CIO should have the support of the people for whom support is being provided. Due to the nature of this position, it is imperative that the CIO also have the support of the Vice Presidents, and, in the case of MTU, the Deans.
- The CIO should have subordinates to whom routine tasks can be delegated. That is, the CIO should avoid daily administrative (and time-consuming) responsibilities that can be delegated to others, such as making sure that everyone has showed up for work and has kept to their schedule. Why? Well, information systems and computer technology are time-intensive and high-pressure (due to financial and time constraints). Needs are goal-driven and should not be crisis-driven. Reacting rashly in crisis situations often causes further damage. Solutions must be well thought-out and carefully designed, implemented, and evaluated in order to avoid crisis-driven situations. And this requires time.
- The CIO should be able to call meetings of relevant persons in order to bring focus to real and/or potential problems and arrive at a consensus for global (and pertinent local) decisions. In practice, the relevant persons will often want such a problem addressed. The CIO must have considerable expertise in information systems and computer technology in order to mediate such discussions and to separate the "chaff" from the "wheat" in such discussions among "polarized" groups.
- The CIO should be able to review all information systems and computer technology requests in order to determine if the particular goals could be better met with an alternate solution. In practice, no one should spend information systems and computer technology funds without approval of both the CIO and the VP for Finance. Over time a history would develop of what are reasonable and unreasonable requests so that only questionable requests would require extended review and that reasonable requests would not be denied (except for lack of funds).
- The CIO should be able to allocate resources to the accomplishment of relevant goals. That is, once a workable solution and consensus is achieved for a particular

system, the system must be implemented in the most timely and cost-effective manner, taking into account all available existing (in-house) resources. These resources would be part of a combined budget at MTU.

Scenarios

We now establish that,

#5: There are various ways to match authority with responsibility.

The following scenarios attempt to match authority with responsibility (note that the responsibilities and authority discussed above are intended to fit into this framework, especially option B.).

A. Low responsibility, low authority: The CIO controls very little (small budget, if any, no people to help, etc.). Every department controls their own budget and can use resources as desired. The CIO is relegated to providing informal leadership to try to hold things together, but has no authority to do anything significant. If responsibility is raised without raising authority, the CIO will be frustrated. Departments enjoy the freedom, but long for some way to create group and enterprise information systems that have high payoffs, but require centralized direction (for example, Snyder 1992 and Snyder, 1993b). Since the CIO would have other primary responsibilities (such as teaching at MTU), the CIO would be in touch with the reality of the business (educating students) but not have authority to do anything about it. Little can be expected to be accomplished in this situation.

B. Medium responsibility, medium authority. The CIO provides proactive leadership, but has some authority (a sizable budget and qualified people to help). Departments still have their own budgets. A workable solution to the budget problem is to take a certain percentage of the information systems and computer technology budget (say 10% to 20%) and use it to subsidize departmental purchases. That is, if the department gets approval, the purchase is subsidized. The department can, however, use funds to do whatever they desire. This should allow departments do deviate when desired, but encourages them to stay with the overall plan. Since the CIO would have some other responsibilities dealing with the reality of the business (educating students), the direction provided would more likely be oriented to these priorities.

C. High responsibility, high authority: The CIO controls everything, both directing and controlling. The budget is centralized and everything must be approved. Take it or leave it (the traditional centralized control approach). Departments are frustrated by the lack of freedom, but group and enterprise information systems can proceed (in a take it or leave it form) if the CIO is at all inclined. Since all responsibility is administrative, the CIO may lose touch with the reality of the business of the institution (educating students). Priorities would be reflected appropriately and really useful systems might not be attempted (for fear of failure or lack of concern).

In all cases, each of the previous steps must be completed and lack of support from the top dooms any approach.

If I were to give one guideline for insuring that centralized direction and not centralized control is achieved, it would be the following user-interface guideline.

> The default of any user-interface decision should be to make the user interface such that the beginning user is least likely to get into trouble, but to provide a means for experienced users to circumvent such decisions.

For a menu system this means allowing a way to get to the DOS command line. For hardware/software purchases, this means recommending (and subsidizing) supported hardware/software, but not blindly requiring it (if someone wants an Apple MacIntosh, seriously consider supporting that request). From a CIO point of view, do not force your viewpoint on others - you may regret the consequences.

Conclusions

This paper is not intended to be the final word on providing centralized direction without centralized control. However, this paper has attempted to create the following line of reasoning.

#1: Someone needs to be in charge of information systems and computer technology.
#2: Organizational change will be required. Are you ready and willing?
#3: Leadership and management, not just management, are required.
#4: The authority must match the responsibility.
#5: There are various ways to match authority with responsibility.
  A. Low responsibility, low authority
  B. Medium responsibility, medium authority
  C. High responsibility, high authority

References

Gibbs, Mark (March 28, 1994). Net Results column, Network World.

Kroenke, D. & Hatch, R. (1992). Management information systems, 2rd ed. New York: Mitchell McGraw-Hill.

Snyder, R. M. (1992). CLASSY: A system to automate class administration tasks. Proceedings of the 1992 Annual Meeting of the Southeast Region of the Decision Sciences Institute. Savannah, GA.

Snyder, R. M. (1993a). Estimating an academic computing budget. Proceedings of the 7th Annual Computers on Campus National Conference. Myrtle Beach, SC.

Snyder, R. M. (1993b). A system for assessing student progress. Proceedings of the 29th Annual Meeting of the Institute of Management Sciences. Myrtle Beach, SC.

Snyder, R. M. (1994). The noncomputer-checked redundancy problem. Proceedings of the 6th Annual Meeting of the International Academy of Business Disciplines. Pittsburgh, PA.

Sommerville, I. (1989). Software Engineering, 3rd. Ed. Wokingham, England: Addison-Wesley.

Proactive Approaches to Information Systems and Computer Security

Robin M. Snyder
Director of Academic Computing
Associate Professor of Information Systems
The Byrd School of Business
Shenandoah University
1460 University Drive
Winchester VA 2601
703-665-4938

Abstract

Networks provide a way to share resources and information and thereby avoid the costs associated with duplication of resources and information. But there are problems associated with the ability to share information, namely, security. In general, one would like to balance the goal of having the best security possible with the cost of achieving that security. Although there are many articles and books available on the subject, it can be difficult to put together a practical and cost-effective system. This talk will focus on practical and cost-effective proactive approaches to information systems and computer security that address the problems of sharing and security of resources and information.

Introduction

Entire books have been written on the subject of network security (Sawicki, 1992). Instead of summarizing such references, this article will focus on some suggestions for practical and cost-effective proactive approaches to network security. And network security includes avoiding problems, detecting problems, and recovering from problems. Thus backup methods and controlling noncomputer-checked redundancy (Snyder, 1994) are an important part of network security.

Background

The sole purpose of a computer network is to share resources - printers, disk drives, files, and information in general. But security is needed to keep certain information from certain people. Loss of this information, or illegal access to this information, can have catastrophic consequences to the organization. Some form of information hiding is required so that access to information is only to those who have a need to know and that any damage can be detected and repaired quickly and efficiently.

No system can be absolutely secure. In order to be useful, however, individuals must have access and that access must allowed and controlled selectively and, from a practical point of view, that requires some way to manage groups (Snyder, 1994). In a small group of cooperating users, the problem is primarily one of accidental misuse.

In a larger group of (potentially) uncooperating users, the problem is more serious. Once on the Internet, every hacker in the world can try to break into your network.

Each increase in security has an associated price tag and cost. No security at all has a low price and a high cost (due to potentially lost information) while high security has a high price and a high cost (due to high price). The fundamental conclusion is that one should use every security principle for which you are willing and able to pay the associated cost. But remember the opportunity cost. That means, do all the easy things. And remember, when you pay for a capability in terms of time and money, you give up whatever else you could have obtained with that time and money. And resources are finite, not infinite.

A dynamic approach to security is to be contrasted to a static approach to security. By analogy, a static approach to securing a building is to spend considerable resources to protect the building, but not to have a human presence there to react to dynamic situations. Just lock it up and hope no one breaks in. A dynamic approach uses many of the static features, but adds a human presence. In the case of the building, a security guard who routinely patrols the area (hopefully in a somewhat random pattern to avoid being predictable). Now, computer networks are electronic, but the same principles apply. You cannot just install a security system and leave it alone to monitor itself. We now look at some principles and selected examples.

Trade Secrets

One should always assume that any adversary has access to all published information, so it is wise to keep information out of the hands of potential adversaries. Any information that is kept secret will make the adversaries task more difficult, since the adversary cannot limit the search for access methods. Thus, in physical security, randomly changing the guard's path is a step in this direction. Another example, military radars. The frequency range that a radar operates at can be easily determined (from physics) by knowing the approximate size of the waveguide. This information is usually confidential. The actual frequency being used by a military unit at a specific time is, however, usually top secret.

When I had to upgrade a (software) network operating system to a new (hardware) server without downtime, I could not do this and switch the printers in one day. But, by keeping secret the access to the printers, only one person (a sharp student of mine) noticed that (for about 2 days), everyone had access to the laser printers.

Another important example is hiding the SUPERVISOR account. That is, make a user, such as REUBEN, system operator privileges (equivalent to SUPERVISOR), give the SUPERVISOR account a very long password, and create a SYSOP user as a normal user. Since it is very difficult to determine who has SUPERVISOR privileges without having those privileges, an adversary would probably attempt to break into the SYSOP account. If successful, the adversary would only gain normal user privileges. And, if the SYSOP is being used as a normal account a good deal of the time, any adversary would think that the system operator is logged on and watching.

User Roles

In an age of increasingly decentralized information systems and computer technology, users must become accept some of the responsibility for the security of their own information. The difference is one of cost and effectiveness. It is generally more cost effective for someone needing medical assistance to use an outpatient service rather than 24-hour intensive care service. Now, when you need it, you need it. But users must come part way and not expect the MIS staff to provide 24-hour intensive care service.

An example is the encryption of files. That is, instead of relying solely on network security, the user can take a proactive approach by encrypting files, such as sensitive files containing contract information, budget information, etc. Most word processors and spreadsheets provide this function (although some, such as WordPerfect, call encrypting locking a file). If not supported, a program such as PKZIP can both compress and encrypt a file.

Remember, though, there are ways for others to unencrypt the file if they get access to it. By encrypting the file, you are just making it more difficult for someone who gets access to your information to decode it. Also, sensitive information can be more secure if stored on a local hard drive (and encrypted). Peer to peer networks (such as Windows for Workgroups) can make it easier for a small group to share information that is not to be stored on the centralized file server.

It is the responsibility of the MIS staff to inform and educate users on these methods. But the user has the ultimate responsibility (for information for which they are responsible).

As another example, consider a traditional monolithic (or smart) backup system where everything is backed up and hopefully, anything lost can be restored. A more proactive approach to information security (at least at the personal level) is to create, at periodic intervals, a snapshot of all of the users files. Then, one a month, provide each user with a printed (or electronic) copy of all changes to their files, all login's, workstations used, etc. during this period. It is then the user's responsibility to identify suspicious activity that might relate to their own information. Such a system is currently in the process of being implemented.

Passwords

Passwords are intended to authenticate that the person using an account is, in fact, the person the person claims to be. It is best to have one account for every user and require a password for each account (some users could have more than one account). If there are more than one user per account, then if something happens, there is no one person responsible. Note: If executive (University-wide) information system support is not available, it may be necessary to permit a generic userid of STUDENT that does not have a password. Such an account cannot have access to email, for the obvious reason that, if something goes wrong, no one is responsible. And who is email to STUDENT delivered to?

Some network systems permit the supervisor to run a security check to see who might have picked a poor password (or none at all). I found it useful to create a program that creates randomly generated, but pronounceable passwords, to be displayed to the user when changing a password. This avoids the problem, "Well, I could not think of a password.". Since dictionary words are easy to guess, I suggest taking a dictionary word, and changing, adding, and/or deleting characters to make it difficult to guess. I also use a simple, but secret, method of generating initial passwords to facilitate the automatic generation of user accounts.

Suppose that you do have executive information systems support so that you can automatically assign userid's and passwords. One institution I was at used a "W" and the last 7 digits of the student's social security number as a userid (and initial password). The advertised policy for default passwords for the fall semester was to use the social security number. This was not a good idea. First of all, 7 digits were already in the userid which was publicly available. Second, the first two digits are not assigned randomly, but are assigned based on place of birth. As most of our students were from the same area, most of the initial 2 digits were the same. Of 42 students I used as a sample, the following distribution applies (X represents a student).
00 X
...
14 XX
15 X
16 XXXX
17 XXXXXX
18 XXXXXXXXX
19 XXXXXXX
20 XXXXXX
21 XXX
21 X
22
...
42 X
...
51 X
...

This advertised policy would have made it very easy to break into the system. Instead of guessing 50 times, on average (2 digits out of 100), only 3 to 4 tries, on average, would have been needed. This was not a very smart policy. From a security point of view:

- 7 digits of a student's userid are currently available.
- The distribution of the first 2 digits is easy to guess.
- You must assume any adversary has access to published information (and will use it).

In summary,

- the policy violates Federal law (Buckley Amendment),
- the current policy is inconvenient for everyone, and
- the policy for initial passwords is an open invitation to breaches of security.

So, I suggested that, instead of using the social security number directly, the default password for user accounts should be determined by a hashing algorithm on the user's social security number. The algorithm should be "computationally expensive, taking on the order of 1 second of computer time, to make guessing by brute force difficult" (and, or course, secret). The key idea here is that any computer program that attempts to guess passwords by off-line brute force guessing would have difficulty. With this guidance, the computer center came up with 32 character case-sensitive passwords (later 24, then 16, hopefully less by now). One mistake and you have to do it over. The irony here is that they misinterpreted the guidance so that they made it take the user over 1 second to type the password, and not a computer attempting to generate passwords (from a known algorithm).

Most networks have a lockout feature to handle guessing of passwords. That is, if an incorrect password is supplied more than a specified number of times over a specified amount of time, the account is locked out for a specified amount of time. The purpose, again, is to hinder brute force guessing of passwords. Reports of lockouts also indicate either an attempted break-in, or a user who is not sure how to use the account. For these reasons, I set the lockout so that if there are 3 bad login attempts within 20 minutes, that account is locked out for 10 minutes. So if a user calls and says that they are locked out, it usually takes them 10 minutes to call, so I tell them to try again. Saves me a lot of work and helps avoid brute force guessing of passwords.

Hard Drives

The information on the hard disks in the a computer lab can be considered a database. Therefore, all the concepts from data base information theory apply. In particular, data consistency of files on hard disks in a PC lab is important for the following reasons.

- Legal reasons: Unlicensed software must not be allowed on computers in the lab. Such practices set a bad example to students using the lab and open the institution to expensive lawsuits should an audit occur.
- Security reasons: Software that might contain viruses must not be allowed in the lab. This includes game programs, a favorite "Trojan horse" method of transporting viruses from one computer to another. At least one virus, the "Falling letter" virus, was found on a hard disk in the lab.
- Educational reasons: Students, especially beginning students, should see the same environment on whatever machine they use. Operators diagnosing problems must be able to quickly identify changes to the default hard disk environment.

Only licensed software should appear on machines in the lab. Computer games should be strictly controlled (games are a primary access method for computer viruses). Hard disks in the microcomputer labs should be periodically updated to conform to a consistent, legal, and safe data environment as determined by the master directory structure.

Accordingly, I have wrote a program, DIRT.EXE, that could be used to assist in maintaining data integrity. This program allows the lab supervisor to

- make a master copy of the approved directory(s),
- erase all files in the lab that are not on the approved directory, and
- identify those missing and altered files (makes it easier for lab attendants to identify problems).

Note that this concept is just as valid and important in a networked environment as it is in a non-networked environment. In the networked environment, one does not have to physically move the disk from workstation to workstation, and the updates can be made automatically. If one has a portable hard drive connecting to the parallel port, the updates can also be made.

This program was delivered in early March 1991 to automate many of the tasks of maintaining a consistent data environment on hard disks in a lab. Documentation on the use of the program was included and a request that the program be used in the lab was made. In late April, I asked why a consistent data file environment was not being maintained on the hard disks in the lab. The person in charge responded that there was not a problem and that the hard disks would be reformatted after the end of the semester.

I decided to do an experiment to determine if, in fact, this was true. Late Friday afternoon (May 3, 1991) and on Sunday afternoon (May 5, 1991), I ran the DIRT program on 18 of 21 computers in the lab (3 computers were down, or results could not be obtained due to disk write errors while saving the results). The results are as follows.

total files	average files	
3265	181.4	extra files (that should not be in the lab)
1119	62.2	missing files (that should have been in the lab)
693	38.5	changed files (that have been altered in some way)

In any experiment, random and systematic errors should be distinguished. A large enough sample size will cause random errors to cancel out. Systematic errors, however, are consistent errors that bias the results in a systematic manner. The systematic errors that were identified include:

- WordPerfect 5.1 was illegally installed on many of the computers, accounting for 97 extra files per computer (where illegally installed).
- Some computers were cleaned the week before in verifying the correct operation of the DIRT program.
- Some files are only needed on the faster (386) machines or can only fit on the machines with larger hard disks. This would increase the number of missing files on the older machines.
- Some files contain state information for programs that need to pick up where they

> left off. These files are updated every time the program is run. This would increase the number of changed files on every computer.

A total of 3265 extra (unauthorized) files on 18 computers in the lab seemed to indicate a problem, contrary to what I was told. In his book, Bently (1982, p. 32) writes

> Programmers are usually notoriously bad at guessing which parts of the code are the primary consumers of the resources. It is all too common for a programmer to modify a piece of code expecting to see a huge time savings and then find that it makes no difference at all because the code was rarely executed.

By analogy, most computer lab administrators do have little idea where inefficiencies are occurring in their labs unless experiments and monitoring are done to identify and quantify the inefficiencies and expected benefits of any policies (such experiments, however, make excellent learning tools and projects for students in computer and computer-related fields).

Conclusions

This paper has attempted to focus on some practical and cost-effective proactive approaches to information systems and computer security that address the problems of sharing and security of resources and information.

References

Bently, J. (1982). Writing efficient programs. Englewood Cliffs, NJ: Prentice-Hall.

Sawicki, E. (1992). LAN desktop guide to security. Carmel, IN: SAMS Publishing.

Snyder, R. M. (1994). The noncomputer-checked redundancy problem. Proceedings of the 6th Annual Meeting of the International Academy of Business Disciplines. Pittsburgh, PA.

Snyder, R. M. (unpublished, to appear). Managing groups in a networked environment.

Computerizing on a Shoe String: With Lots of Prayer

Richard N. Stewart
Assistant Professor, Church Administration
Director of Field Education
Media Center Coordinator
Lutheran Theological Seminary at Philadelphia
Philadelphia PA 19119
215–248–4616
richard_stewart.parti@ecunet.org

Abstract

This paper follows the use of computers since 1984 and the five year relationship that ASCUE has had with the Lutheran Theological Seminary, through Rev. Richard Stewart. The overall effect is to assist a truly small school move from backing into computerization by use of word processing by support staff and data base management by the Development Office, to developing a formal plan for the use of technology for the 21st Century. Central to this turn around was a developing administrative awareness. The administrative encouragement has assisted the Media Center move toward developing a strategic plan for technological development for the seminary.

Background

As early as 1983, the Lutheran Theological Seminary at Philadelphia began the search for a way to effectively use computers. With the services of EDUCOM and consultant David Todd, Director of University Computing for Wesleyan University, recommendations for the computerization of the seminary were received. The forty–four page report, though daunting was taken by faculty members as a starting point for planning their personal and professional use. The development office knew that it had a problem in keeping track of the alumni and the donors to the seminary. Computerization was something that everyone in education seemed to be talking about.

The recommendation for the seminary was that they purchase an NCR Decision Mate V with the ?optional? 8088 processor to develop a local area network. The network was to be a Omninet System from Corvus Computers, with a NCR Modus central file server with 32 M bytes of hard disk storage. The primary institutional functions were to be word processing, spreadsheet analysis and database management. Computer Aided Instruction was mentioned with the recommendation going to Apple Computers.

Faculty were to be encouraged to purchase their own computers so that their writing and lecture preparation could be facilitated. Faded memories will recall the ?compatibility? achieved by names that were being considered at the time for faculty purchase, such as Calico/Adam, K–Pro, Compaq, Chameleon, Eagle, and Tandy 2000. Though Word Star or Word Perfect was the recommended word processing software from consultant Todd, the seminary chose Microsoft Word as a standard. Recommended suppliers were Sears Business Center and Jonathan?s.

As with any institution, they sought the advice of a second ?expert?, John Schmitt, the General Manager of Systems and Computer Technology Corporation. Hisbusiness provided consulting services on installation of large main frame systems in large college and university settings. His advice was synthesized by the faculty committee to six pieces of advice:

Proceed slowly with computerization
Go IBM
No Networking now but purchase equipment with the capability
Software decisions should be made before hardware. Thus it may be practical to bring to bring in software specialists before sales persons.
IBM PC Juniors for home use would be compatible with the IBM PC?s at the Seminary
Technical staff will be required

The faculty set out to take the recommendations of Mr. Todd and Mr. Schmitt and make them their own. With a report on January 3, 1984, the proclamation was made that the jury had reached a verdict. The seminary was going IBM. That proclamation was tempered by the fact that the recommendation that the institution assist the faculty in their purchase was a ?waterless? cloud. Note was taken of the potential cost savings of ?compatible? computers, but even those costs exceeded $2,000.

A third consultant, a Lutheran chair of a data processing department at a Central Pennsylvania business school, raised questions about the maintenance of the communication lines for a network, the training of staff for various pieces of software and the maintenance of equipment. He suggested carry–in maintenance, the training of current staff to be consultants to one another, and raised the issue, again of computer assisted teaching.

Floundering in this sea of information the administration went to a trusted contact, their accounting firm. They agreed to advise and assist in the purchase of equipment and design and maintain database software for the development department and accounting software for the business office. After some deliberation and pressure from the support staff who were becoming increasingly overwhelmed. The seminary?s accounting firm Mathieson, Aitken and Co. were contracted to set up the data base function for the seminary, as that seem the most pressing area for the long awaited computerization.

In 1984, they provided a data base, Metafile, with a report writer, and a word processor. These pieces of software were first installed on the computer purchased for the Development Office secretary, with the software costing about $8,000 and the hardware about $10,000. The same software was modified for the secretary in the Field Education Office by the end of 1986.

Simultaneously the library was being connected with OCLC and beginning to share information with local libraries through the Philadelphia Area Libraries Information

Network. (PALINET). All of their computers were for library staff – librarian, and assistant librarian and the beginning of catalogue computerization. Library patrons saw no direct benefit from the computerization of the library staff.

With grants from the Prosser Foundation in early 1985, the faculty were able to obtain an initial grant of $500 to assist with the purchase of a personal computer. While they were encouraged to purchase IBM computers through the contact with Mathieson & Aitken, the grants would be available for any computer that was IBM compatible. Training was provided by Mathieson and Aitken. Most of the faculty limited their computer use to word processing and personal finances. These facultypurchases also spurred on the purchase of a computer for the faculty secretary.

Each support staff person approached this new office tool with a bit of fear. Yet at each stage of progress toward computerization, there was one staff person who had a vision of what the future might bring. The Media Center Director, Jeff Davis, was the first of these visionaries. He was the resource person for Rbase and DBII. He also had a personal interest in WordStar. With his departure in 1987, the new Development Director, Richard Husfloen acquired a reputation for buying the next hot item. He raised the consciousness level of all the staff to the potential of computers. He facilitated the standardization of Paradox as the institutional database, as the Metafile system had become unwieldy. As the report generator was at best difficult, it made support staff dependent upon the developer for service calls to complete their tasks. Husfloen also encouraged a regularized schedule of backup. His office was the first to make the transition from 5–1/4 disks to 3.5 floppies.

With his fundraising efforts, the reality of a matching gifts from IBM came in 1989. January 1989 also marked my arrival on the faculty. At that time my claim to fame was the inability of saying no when someone said that something was wrong with their computer. I seemed to be one of the few that was not afraid of opening up a computer and looking in, sometimes being fortunate enough to get it running again.

Introduction of an ASCUE Influence

In June of 1990, I attended my first ASCUE Conference. Besides being a neophyte to academic conferences, I also considered my knowledge of computers to be minimal. I came with lots of questions and no forum for finding answers. In a nine month period prior to my first ASCUE Conference, a campus computerization committee had struggled trying to determine how to spend the matching grant gift from IBM and employees. With a majority of the administrative software at that time designed for mini-computers, the UNIX operating system became a topic of discussion. While that gave the greatest amount of accessibility to administrative computing, 9 months of discussions hinged around one issue, who would be the system administrator. As I was new to the faculty, no one wanted to place that responsibility on someone who was both unproved and not certain of continuation. The director of development, though a champion for computerization, defined himself out of the consideration. The assistant registrar didn?t even consider a request, as he felt that his multiple responsibilities would not leave him with enough time for the current demands he faced. The final decision was to purchase stand alone computers. There would be no network at this time.

ASCUE provided me an opportunity to listen to a host of others, who have multiple responsibilities, and significant experience making the same kinds of decisions that my seminary?s administrative staff were struggling to understand. Of significance for me were the vendors who displayed Administrative software. All of their materials were of important, as I shared my learning with the various departments who were computerized or were being considered for computerization. Also at that first conference, I won one of the giveaways. At the time I had hoped for one of the pieces of software, but instead I won a subscription to EDUTECH. At first glance it seemed to be beyond my level of competence, but as the year went on, the articles in each issue took on greater importance as questions were raised about the prospects for full administrative computing.

The institution had computerization in place for the following areas:
Faculty
Development
Admissions
Support Staff
Field Education
Admissions
Faculty Secretary
Development

There are some obvious absent areas in need of support. You might notice that I have avoided mentioning the business office in this entire scenario. While the business manager had recently purchased an Apple computer for his home, he was concerned about the security of data and the difficulty of a transition from his current system to a computerized one. The bookkeeper flatly stated that when they got a computer, she would quit. To this date their office is the only office that has no computers. Much of what I learned at that second ASCUE conference had to deal with how institutions became computerized. Generally the starting point was in the Business Office with development, and then other departments not far behind. Though we started with a call from the development office, we seemed to be backing into computerization missing the link that seemed to united other institutions. While we did not have to overcome mainframe mindsets, we did/do have to overcome mindsets.

Providing information for the new registrar about administrative software was a much easier task. I came with an agenda to look at Administrative Software to assist the new registrar in his transition. He was replacing a legend who did not need a computer, he was one. What he didn?t know by memory, was readily at his finger tips. They had given the new registrar a year to be an apprentice before the pending retirement. With the retirement of the President?s Secretary, another opening for computerization appeared.

ASCUE II (1991)

During the next year, I went through a six month period of chemotherapy. Along with that, I organized a program to obtain computers for our student labs. We upgraded all word processing programs to the latest version of Microsoft Word. We began to explore the benefits of communication software for communicating with church offices and other on-line information services. I was/am an active member of the Ecumenical Network (ECUNET). My second ASCUE Conference was a time of recuperation as it was the first trip after intensive medical supervision. Though we had not chosen the option of a network for administrative hardware, the need was still ever present for administrative software in our internal discussions.

ASCUE III (1992)

By the time I came to my third ASCUE Conference, every staff support person was using a computer, either new or one that had been passed down after being replaced for greater functionality. Increasingly I became the focal point for maintenance and updating of hardware and software. With the responsibility came little authority. Using the Proceedings of previous conferences, I began to share images of what itmeant to have a long range plan for technological development. These documents were shared with the Librarian and summaries were shared with the Dean and the President. Recommendations for purchases came from my office, with the authority to make purchases still lodged in the person of the director of development, with whom I had a good working relationship. But his departure for another position, away from our campus, threw the whole role of computer advising into the shadows.

Articles from the EDUTECH journal helped to shape lines of authority for developing student staff to support up and running computers. The journal was being renewed at the request of the Dean who felt that the readings were helpful for someone who did not have time to keep up with the technology. It provided him when requested with authoritative direction. Again Administrative software was an important issue for me to obtain additional information. We had begun the process of seeking foundation funding for the one time costs involved in bringing every administrative office on line, including the business office. The information garnered at that ASCUE helped in putting a proposal together that would serve Admissions, Registrar, Development, and the Business Office. We also sought funding for the equipment, software and staff time for the library. At that time and even yet today we have not found adequate library software that has been developed outside of library resources. Maintenance of equipment had begun to be a problem, but we found that several of our students had a variety of computer experience ranging from computer sales personnel to staff at Bell Laboratories.

ASCUE IV (1993)

For the first time since my first ASCUE Conference, I began to focus on the educational uses of computers in the classroom. For my personal use, the exposure to several different versions of presentation software captured my imagination and added fuel to the practical aspects of my graduate study in telecommunication and distance learning. I still struggle with the issues of small school, small budget, and increasing expectations. With increasing technological possibilities and costly equipment and software small schools find that budgets can no longer contain the full breath of expectations.

While waiting for the foundation to make a decision on the administrative computerization proposal, the workload continued with the purchase of two laptops for on road record keeping and presentations. A LCD panel was purchased for classroom use, and used for the first time with the American Bible Society interactive Biblical translation. All support staff with one exception have computers with CPUs of 286 or faster. All computers are using Microsoft Word 6. Backups are being done with regularity. Pirate copies of software are being removed. Educational purchasing of software is being encouraged among faculty and students. Support is provided for Paradox and Alpha 4 and Word Perfect Presentation 2.0.

New areas of interest have emerged with the potential of desktop video and computerized editing of video work at the seminary. With the emerging synthesis of computer technology and media, our media center really is starting to encompass a multitude of media formats with the computer taking center stage for them all. The recent announcement of the $125,000 foundation grant also predicts a good deal of work over the coming summer and an attempt to creatively use the new information that will undoubtedly come from ASCUE 1994.

Future Challenges

While the administrative computer task force has narrowed the choices of software to about three vendors, the challenge still faces us that the last office to be computerized will be the business office. We are looking at a 9–12 month phase in period. The business manager has decreased his level of apprehension about security and has taken on the task to make sure that the software packages closely resemble his current paper trail. He has generally found that he will be able to report in much more detail. The bookkeeper hasn?t said much recently, but I firmly believe that she too can be a product of conversion.

I am being asked to develop some internal standards for the possibility of video conferencing via computers for continuing education and for teaching students who are unable to physically relocate to the seminary. I will look for leads at ASCUE as we move forward with our use of computers, hopefully now walking forward.

Freshman Biology Majors Experience Authentic Learning by Producing Their Own Lab Manual

Paul S. Tabor
Biology Department
Clarke College
Dubuque IA 52001
319-588-6526
ptabor@keller.clarke.edu

Authentic learning refers to an approach where students learn by interacting with the subject matter in the same way that they will beyond the classroom. For the second semester freshman biology majors at Clarke, this means that students become familiar with the process of scientific investigation and rely on this process to obtain answers to questions of their own design. Primarily from their own experiences, students draw conclusions and recognize the general concepts of the lab exercises as logical extensions of their own findings (often in comments such as "We showed that in our experiment."). The use of computer networking has been integrated into this lab course and is an important element in the success of the authentic learning approach. Students in the Principles of Biology II lab course have experience in the use of the analytical and communication capabilities of the networked, multimedia classroom from the first semester lab course. Students use a range of applications software, a campus e-mail class distribution list as a forum for communication and exchange of information, and the Internet. The evaluation of the performance of students in this lab course is based on the lab groups' contribution to the development of a class lab manual of the exercises accomplished over the semester and individual interim progress reports.

The Approach to Authentic Learning

The objectives of this lab course are 1) to give students the opportunity to become familiar with the process of scientific investigation as an effective, exciting way to learn, 2) provide an opportunity for students to use their own findings to help develop an understanding of relationships between structure and function/process at the organismal level, and 3) provide the scope of biological sciences that is needed for students to make decisions on an area of further concentration in biology. In the initial lab sessions, student teams are formed and the teams begin the task of deciding on the questions that they want to investigate. The questions must relate to the six broad topics that need to be studied during the semester. Initially, students conceive two lab experiments. To aid the students, resources called "readings" that are condensed reports of research studies are made available to students as files on the ELECT classroom network. There are from four to ten readings for each topicarea. During the first two weeks of class, students read and discuss the readings. The discussion, for the most part, occurs via the class e-mail distribution list that includes students and the instructor. In class, students work to reach a consensus and then define the first two experiments for the semester. Thus, from the beginning, the lab groups must make decisions that will affect the rest of their semester -selecting the lab experiments they wish to do. This process leads to writing the introduction sections of the lab exercises, including the background and the hypothesis statements, outside of the lab session. The students usually choose the option of assigning themselves various parts of the introduction rather than each student writing a complete introduction. Well-written introductions with well-stated hypotheses are very important because the students use the introduction as a guide for writing the conclusions section. In the first two lab sessions, the lab groups design the methods to be used and identify the materials to be obtained. They set schedules for the experiment and assign group members to be responsible for specific activities. Each student contributes his/her own section to the lab exercise's materials and methods section. Although some of the lab experiments appear to be mammoth undertakings for freshman, the process is broken down into segments and the students are not overwhelmed.

While some longer experiments are being done, e.g., role of various light regimes in plant growth, development and reproduction, the experiments to follow are chosen and defined. As a result, students are contributing to the introduction and materials and methods of one experiment and analyzing results of another experiment. Since the experiments are new to students and the instructor, all of the students have essentially equal backgrounds for predicting the final outcome during the course of the experiment. The students with the best organizational skills are often leading their groups during the labs. The students who learn easily from graphic representations of results are often the ones to arrive at the conclusions first. These students are not necessarily the ones the produced the graphs or figures.

The first experiment this semester included many independent measurements (growth, transpiration, direct fluorometric measurements of chlorophyll concentrations in whole leaves) that gave students the opportunity to do multiple variant analysis and provided experience in evaluation of the quality of data. The students chose this experiment partly because one of the readings was a related study from a senior thesis of a recent Clarke graduate. The second topic chosen was animal reproduction and development. Students became interested in the readings describing birth defects and the causes of birth defects. As an investigative experience, the class decided to attend genetic counseling sessions between families and physicians at a regional hospital. In this case, students found it necessary to locate specific resources discussing the reproductive and developmental problems they had observed first hand. The students used the Internet as a tool to find this information on a need-to-know-basis. The third exercise was the comparison of circulatory systems of deuterostomes and protostomes. The experiments included latex injections and dissections, and cardiac output determinations. Observations were video recorded and digitized full-motion clips are included in the lab manual in the results section of this multimedia lab exercise. The multimedia production process required the students to carefully analyze their recorded observations.

At the time when the results of each exercise are being analyzed by the groups, additional readings, such as review articles which discuss the concepts addressed by the lab experiments, are made available on the network. The instructor initiates discussions in class that continue on the e–mail distribution list. The purpose of the review articles is to stimulate the discussion of the important concepts by generalizing from the findings of the students' experiments. Understanding is aided by the fact the students have direct interaction with the subject. It is common for the e–mail discussions to evoke other related concepts from the students.

In past semesters, students submitted their work as files to a "handins" directory on the network server. Students were not able to edit the material once submitted and would have to resubmit the whole file with changes. The files in "handins" had to be copied by the instructor to individual "handouts" directories for students to copy files to their own disks for editing. This semester, the campus e–mail network has become the mechanism for communication and information exchange. Using Pegasus Mail in Windows, students submit their work as attached files to e–mail messages. The files received by students need to be transferred to disks and then opened in the appropriate application software before they can be edited. The files can be sent to any student or students or to the whole class via the class distribution list. The distribution list makes it possible for any or all students to make editions to any submitted sections of the lab manual. The poorer writers are able to see editing by their classmates on the preliminary versions of the lab manual. The best writers are encouraged to be critical of their own reports as well.

Results of the Authentic Learning Approach

There are clear differences in the activities of students in Principles of Biology II lab as a result of the authentic learning approach. Many of the changes have brought the experiences of the freshmen in biology closer to the experiences of research scientists. From the beginning of the semester, the students need to work collaboratively. The decisions are made by consensus of their colleagues, schedules are set and conflicts are resolved by collaborative group efforts.

The networked computer classroom provides a unique collaborativelearning environment. For example, as a student group analyzes results and produces a graph, the other groups crowd around the workstation to learn the technique. The groups then return to their own workstations to implement the new technique themselves. The e–mail distribution list was a novel forum for the lab class to seek and exchange information. Students were much less hesitant to seek help by asking questions of each other via e–mail. Students are also less hesitant to suggest that other students should make more effort to contribute. The students learn that they are dependent on other members of the group for their sections of the lab manual as well as responsible to the group for their part. Although the investigative approach in the lab provides an opportunity for in–depth study so students can better relate to the concepts, the communication between students via e–mail was just as important for building an understanding of general biological principles. Many details, especially on methodology and information on materials specifications, that were needed by students were found on a need–to–know basis using Internet resources.

Building the lab manual resulted in a change in students' attitudes towards the work in the lab and provided the students with a sense of accomplishment. The students began to see that each exercise that they finished became a product. The students developed an appreciation for the process of scientific investigations. The attitudes changed from ""this is going to take a long time" to "we need to get this done because it is our plan". It is notable that the technology used by the students saved them a great amount of time in their collaborative tasks and in distributing the information to the class.

In this authentic learning approach, the instructor needs to lay good groundwork and then stand back for the most part. The groundwork consists of providing readings for the students at the beginning of the course. Then providing some technical literature for methodology and its background. Developing a focused hypothesis for each lab exercise is important and the instructor must give students some guidance without contributing. The students are always interested to know that the instructor is interested in what they are doing. This is best accomplished by the instructor contributing to the discussions on the e–mail distribution list forum.

The most appropriate way to evaluate the students is to contribute to the discussions on the distribution list and observe who is active and comment to students individually about their participation relative to other lab group members. If students know that participation contributes to the evaluation, they quickly look to see how much others are contributing.

The authentic learning experience of freshman biology majors in Principles of Biology II lab was enhanced because of the e–mail forum for communication and information exchange and the analytical capabilities of the networked computer classroom. This will be important for the success of our students because beyond the classroom students will utilize these technologies in their interactions with process of scientific investigation.

Stimulating Simulations

Ken Taylor
Thomas More College
333 Thomas More parkway
Crestview Hills KY 41077
606-344-3417

Instructors in computer programming classes, in addition to teaching the syntax of the language, are always searching for programming projects to assign their students. These projects, for the most part, should be both challenging and interesting thus motivating the student to perform to the best of his ability. The field of computer simulation provides a rich assortment of such projects and in the pages that follow you will be provided with seven problems whose solutions can be obtained merely by simulating random numbers. While it is also true that the problems can be solved mathematically, some students may not possess the required background. This is the beauty of simulation. The mathematics required is minimal. Most of the solutions will utilize arrays thus giving the students valuable experience at using one of the most effective tools at the disposal of the programmer.

All programs are written using the C Language but the projects themselves would be appropriate for any programming language. A listing of all programs may be found at the end of this paper. All that is required is a random number generator. While it is true that random number generators are provided with most compilers, the decision was made to use the one proposed by D.H. Lehmer and implemented by Park and Miller.[1] This random number generator generates uniform random numbers on the interval [0,1) but for the problems that follow, random integers will be required. This can be accomplished in the following manner. let U be the upper limit and L be the lower limit of the integers required. Then the expression

```
(int)((U - L + 1)*rnd) + L
```

where rnd is the uniform number generated, will give integers on the interval [L,U]. For example, if U = 85 and L = 23, the expression

```
(int)(63*rnd) + 23
```

will generate integers between 23 and 85 inclusive. Let us now proceed to the first example.

Keys In A Jar

A contest is run in the following manner. 100 keys, one of which starts a brand new 1994 Lexus automobile, are placed in a jar. One hundred contestants select keys, one at a time, and try them in the car. If the car starts, the individual wins the car. If not, the key is discarded and the next contestant picks a key. Since there are 100 contestants and 100 keys, someone will eventually win the car. Is any one position for picking the key any more advantageous than any other position?

The answer to this question can be obtained as follows. Think of the keys as being numbered 1 - 100. A random integer can be generated in this range and this would represent the winning key. Next let picks[] be a one dimensional array of length 100. A second random number in [1,100] is now generated and placed in picks[1]. This represents the key picked by the first contestant. A check is then made to see if that individual won the contest and if so, it is over. If not a second random number on [1,100] is generated and placed in picks[2] thus representing the second contestant's key. The process is continued in the same manner for each contestant until there is a winner. Care must be taken to insure that the same number is not generated twice since keys are discarded when there is no winner. This is why an array is used. As each number is generated, the array can be searched to be sure that the number has not been previously used. If so it is simply replaced by another random integer on [1,100].

Such a contest can be simulated a large number of times and a record can be kept of how many times the contest is won by each individual person. The results of 10000 simulations are listed in table 1. As you can see, no position appears to have an advantage over any other position. Each contestant won approximately 1 percent of the time. This conclusion can be confirmed mathematically. Let Ai represent the event that the ith person wins the contest. Then

$$P(A1) = 1/100$$

$$P(A2) = P(A1' \cap A2) = (99/100)(1/99) = 1/100$$

This is true since, in order for the second person to win, the first person must lose. Similarly

$$P(A3) = P(A1' \cap A2' \cap A3) = (99/100)(98/99)(1/98) = 1/100$$

In the same manner, it can be seen that each individual has a 1/100 chance of winning regardless as to where he picks his key. Try convincing the person who picks last that he is just as likely to win as anyone else.

Keys in a Jar

# Simulations = 1000

Position	No Wins	Position	No Wins
1	92	51	97
2	75	52	103
3	97	53	82
4	95	54	101
5	100	55	114
6	100	56	90
7	89	57	96
8	105	58	106
9	78	59	102
10	96	60	106
11	107	61	124
12	104	62	108
13	93	63	98
14	115	64	105

Position	No Wins	Position	No Wins
15	95	65	83
16	97	66	104
17	116	67	102
18	97	68	103
19	104	69	121
20	92	70	99
21	98	71	91
22	103	72	103
23	97	73	82
24	114	74	84
25	112	75	111
26	96	76	85
27	104	77	106
28	120	78	108
29	98	79	101
30	95	80	99
31	112	81	104
32	105	82	104
33	90	83	82
34	88	84	93
35	107	85	111
36	106	86	106
37	93	87	103
38	107	88	109
39	106	89	90
40	84	90	105
41	98	91	79
42	103	92	92
43	103	93	117
44	94	94	118
45	105	95	95
46	109	96	96
47	97	97	88
48	101	98	109
49	113	99	84
50	91	100	105

Table 1

Dizzy Dice

The next problem is one that is not intuitively obvious yet still solvable using simulation. Which is more likely, rolling at least 1 six in 4 rolls of a single die, or rolling at least 1 twelve in 24 rolls of a pair of dice? While this could be approximated by actually rolling the dice, it would take a large number of rolls and can become quite boring. Simulation is the more obvious approach. A random integer can be generated on [1,6] thus representing a roll of a single die. This can be repeated 4 times thus simulating 4 rolls of the die. A check is then made to see if a six is obtained. Repeat for a large number of times and keep track of the percentage of times a six is obtained. As for the rolls of a pair of dice, simply generate two random integers on [1,6], add the results and see if it is a twelve. This is then repeated 24 times simulating the 24 rolls. The entire process is repeated a large number of times and the percent of twelves can be calculated. The results of 100000 simulations revealed at least one six in 4 rolls occurred 51.836 percent of the time while at least one twelve on 24 rolls occurred 49.209 percent of the times. These results are consistent with the theoretical values calculated below.

P(no 6 in 1 roll) = 5/6
P(no 6 in 4 rolls) = (5/6)4
P(at least 1 six) = 1 - (5/6)4 = 0.517747

P(no 12 in 1 roll of a pair of dice) = 35/36
P(no 12 in 24 rolls of a pair of dice) = (35/36)24
P(at least one 12) = 1 - (35/36)24 = 0.491404

Chase The Ace

The next problem deals with playing cards. If a regular 52-card deck of playing cards is well shuffled and cards are turned over one at a time until an ace is encountered, on the average, how many cards will be turned over? This simulation also utilizes arrays. Let deck[] be a one dimensional array of length 52 representing a deck of playing cards. For purposes of this simulation, we can simply fill the array with the positive integers from 1 to 52. Since all we are really concerned with is the aces, any number congruent to 1 mod 13 will be considered an ace. Thus the numbers 1, 14, 27 and 40 will represent the aces. The deck is shuffled by making a pass through the array and at each position i generate a random integer j on [1,52] and then swapping the contents of deck[i] and deck[j]. After the deck is shuffled, make a second pass through the array and find the position of the first ace. Repeat this a large number of times and then calculate the average number of cards turned over. The results of various numbers of simulations can be seen in table 2. At this point, one might ask the question "Just how accurate are these results?". The theoretical solution is obtained in the following manner.

Let A_i = 1st ace appears on the ith card

Then

$$P(A1) = 4/52$$
$$P(A2) = (48/52)(4/51) = (48/51)*P(A1)$$
$$P(A3) = (48/52)(47/51)(4/50) = (47/50)*P(A3)$$
$$P(A4) = (48/52)(47/51)(46/40)(4/49) = (46/49)*P(A3)$$

In general

$$P(Ai) = ((50-i)/(53-i))*P(Ai-1)$$

The expected value can then be calculated

$$E = \sum (i * P(Ai)) = 10.600001$$

As you can see, the simulated value is somewhat off for small n but as the number of simulations increases, the simulated value begins to approach the theoretical value.

# Simulations	Average # Cards To Get 1st Ace
100	12.430000
1000	10.796000
10000	10.708800
50000	10.615240
100000	10.585780

Table 2

Hat Check Problem

The next problem is a classical one and can be found in Kemeny, Snell & Thompson. [2] N men, all wearing hats, check their hats upon entering a restaurant. The hat check person gets them all mixed up and returns them at random to the N men. How likely is it that no man gets his own hat back? This problem also has a hidden feature.

"Does the solution depend on the number of men?" At first glance it would seem that it would. This solution also lends its self well to arrays. Let hat[] be a one dimensional array of length 52 and fill the array with consecutive integers from 1 to 52. These integers can represent numbers on the hats. Make a pass through the array and shuffle the numbers as you would a deck of cards. Then make another pass through the array to see if hat[i] = i for any i. If so that individual got his own hat back. If not, no one got their own hat back. The results of the simulation are listed in table 3. As you can see the answer is approximately .36 and the results appear to be independent of the number of men. In fact, Kemeny, Thompson and Snell point out that the theoretical solution approaches 1/e which is .367879..., as the number of men increases. The simulation bears this out.

Simulation of Hat Check Problem

# Men	Prob. at Least 1 Gets His Own Hat	Prob. no one gets His Own Hat
2	0.503600	0.496400
3	0.672900	0.327100
4	0.631100	0.368900
5	0.633900	0.366100
6	0.632100	0.367900
7	0.630500	0.369500
8	0.637500	0.362500
9	0.631900	0.368100
10	0.631900	0.368100
11	0.640300	0.359900
12	0.633100	0.366900
13	0.630000	0.370000
14	0.634900	0.365100
15	0.636300	0.363700
16	0.634100	0.365900
17	0.638900	0.361100
18	0.635200	0.364800
19	0.637900	0.362100
20	0.630100	0.369900
21	0.632100	0.367900
22	0.633500	0.366500
23	0.629600	0.370400
24	0.629100	0.370900
25	0.631400	0.368600
26	0.630900	0.369100
27	0.632200	0.367800
28	0.634600	0.365400
29	0.640600	0.359400
30	0.633600	0.366400
31	0.630200	0.368900
32	0.636100	0.363900
33	0.632200	0.367800
34	0.634000	0.366000
35	0.628700	0.371300
36	0.630700	0.369300
37	0.628900	0.371100
38	0.631200	0.368800
39	0.638600	0.361400
40	0.640400	0.359600
41	0.628100	0.371900
42	0.631700	0.368300
43	0.631900	0.368100
44	0.631900	0.368100
45	0.628500	0.371500
46	0.642000	0.358000
47	0.630500	0.368500
48	0.633500	0.366500
49	0.635600	0.364400
50	0.631700	0.368300

Table 3

Birthday Problem

In a room containing n people, how likely is it that at least two will have the same birthday? By the same birthday, one means the same month and day but not necessarily the same year. This problem is also considered classical and can be found in Kemeny, Snell & Thompson. We take the array approach again. Let b[] be a one dimensional array of maximum length 365. Make a pass through the array and generate random integers on [1,365] , and use them as corresponding array elements. These numbers will then represent birthdays. For example, 1 represents Jan 1, 2 represents Jan 2, 32 represents Feb 1 etc. Assume it is a non leap year. All that remains then is to make passes through the array to see if any two array elements are the same. If so those individual have the same birthday. If not, no two have the same birthday. Simulated results can be seen in table 4. The results are rather surprising to individuals who are encountering this problem for the first time. As you can see, the probability is better than 1/2 for only 23 people in the room and increases rapidly as the number of people in the room increases. Again, though, one might inquire as to the validity of the results. A look into the theoretical probabilities will reveal that the simulated results are right on target. Let

A = At least two people have the same birthday
A' = No two have the same birthday

Also let

B1 = 1st person has a birthday
B2 = 2nd person has a different birthday
B3 = 3rd person has a different birthday (different from 1st two)
B4 = 4th person has a different birthday (different from 1st three)
etc.

Then

P(A') = P(B1 _ B2 _ B2 _ . . . _ Bn)
= (365/365)(364/365)(363/365) ... ((366-n)/365)
= 365!/((366-n-1)!*365n)

And

P(A) = 1 - P(A')

A list of theoretical probabilities can be found in Table 5. As mentioned, they are very close to the simulated results and the average person has no idea that the probabilities are that high.

Simulated Probability of at Least 2 People
Having the Same Birthday

# People	Probability	# People	Probability
2	0.002700	26	0.600500
3	0.008800	27	0.628000
4	0.018100	28	0.667200
5	0.026100	29	0.681700
6	0.041100	30	0.709000
7	0.051300	31	0.731000
8	0.080300	32	0.751400
9	0.100000	33	0.774000
10	0.113500	34	0.798300
11	0.140200	35	0.816100
12	0.158700	36	0.831400
13	0.196000	37	0.852200
14	0.226100	38	0.866700
15	0.255600	39	0.874300
16	0.291600	40	0.890300
17	0.318600	41	0.901000
18	0.348500	42	0.913600
19	0.373600	43	0.925700
20	0.404600	44	0.934900
21	0.443000	45	0.941900
22	0.476700	46	0.948500
23	0.506400	47	0.955900
24	0.548000	48	0.958000
25	0.560600	49	0.968300
		50	0.969900

Table 4

Theoretical Probability of at Least 2 People Having the Same Birthday

# People	Probability	# People	Probability	# People	Probability	# People	Probability
2	0.002740	26	0.598241	13	0.194410	37	0.848734
3	0.008204	27	0.626859	14	0.223103	38	0.864068
4	0.016356	28	0.654462	15	0.252901	39	0.878220
5	0.027136	29	0.680969	16	0.283604	40	0.891232
6	0.040462	30	0.706316	17	0.315008	41	0.903152
7	0.056236	31	0.730455	18	0.346911	42	0.914030
8	0.074335	32	0.753348	19	0.379119	43	0.923923
9	0.094624	33	0.774972	20	0.411438	44	0.932885
10	0.116948	34	0.795317	21	0.443688	45	0.940976
11	0.141141	35	0.814383	22	0.475695	46	0.948253
12	0.167025	36	0.832182	23	0.507297	47	0.954774
				24	0.538344	48	0.960598
				25	0.568700	49	0.965780
						50	0.970374

Table 5

St. Petersburg Paradox

For a fee, an individual may flip a coin until he gets tails. If he flips the coin once, he wins $2, if he flips the coin twice, he wins $4, if he flips the coin three times, he wins $8 etc. In general, if he flips the coin x times, he wins $2x. How much should the fee be to make this a fair game? Since the probability of x flips is (1/2)x, the expected value in this game is infinite.

$$E = 2(1/2) + 4(1/4) + 8(1/8) + 16(1/16) + \ldots$$
$$= 1 + 1 + 1 + 1 + \ldots$$

Consequently, one should be willing to pay any fee to play the game. However, due to time limitations and finances, it generally is not too feasible to pay a high fee since one may run out of time or money before regaining any loses. Thus, given a finite amount of time and money what would be a reasonable fee to play this game? This can be simulated by generating a random sequence of 1's and 0's until a 0 occurs where a 1 represent heads and 0 represents tails. The amount of money won would then be $2x where x is the amount of numbers generated. This can be repeated a number of times and the average winnings calculated. In order for the game to be fair, the fee should then be this average since an individual will neither come out ahead or behind. Results are listed in table 6 and it is interesting to note that no clear answer appears to emerge. Upon reflection, however, this is understandable as the expectation is infinite and consequently anything is possible in a finite amount of time.

St. Petersburg Paradox Table 6

# of Simulations	Average Winnings
100	6.30
1000	11.10
10000	23.59
50000	17.89
100000	17.69
200000	39.45

Progressive Bingo

Many of the cruise ship lines and a number of Bingo establishments offer a special version of the game called Progressive Bingo. In this game, a large amount of money can be won if an individual covers his bingo card in a specified amount of numbers or less. If there is no winner, the game continues until there is a winner but for a lesser amount of money. The next time the game is played, which is generally the next night on a cruise, the amount of numbers is increased by one. How likely is it that this game will be won in the specified amount of numbers?

In Bingo establishments, this procedure continues until there is eventually a winner. However, since most cruises last only seven days and the last day is used for disembarking, the game can only be played a maximum of 6 times. In reality, it is probably only played 5 times since the first day of a cruise is rather hectic and passengers have not yet settled into a daily routine.

A more interesting question would be "On a typical seven day cruise, how likely is it that one will win the game of Progressive Bingo?". Most games start out with 44 numbers as the limit and progress to 48 if the game is played for 5 consecutive nights. This number could possibly increase to 49 or 50 if a game was squeezed in on every possible day.

The answer to this question is ideally suited to computer simulation and the results are quite interesting. To understand the simulation, one must first be familiar with the game of Bingo. A typical Bingo card consists of 5 columns of numbers labeled B, I, N, G and O. The numbers under the letter B are randomly chosen from the numbers 1 through 15. Similarly, the numbers under I are between 16 and 30, under N between 31 and 45, under G between 46 and 60 and under O between 61 and 75. There is also a FREE space in the middle of the card under the letter N. A typical Bingo card might be

B	I	N	G	O
5	21	44	55	72
10	17	32	59	63
1	29	FREE	46	69
13	19	39	48	71
3	22	35	60	61

The first step in the simulation will be to randomly generate a Bingo card. This is accomplished by using a one dimensional array card[] of length 75. The array will be initialized to 0's and as the numbers are generated, the corresponding array elements will be replaced by 1's. Consequently, there is a need to generate 5 random integers between 1 and 15, 5 between 16 and 30, 4 between 31 and 45, 5 between 46 and 60 and 5 between 61 and 75.

The following formula was used to generate all 24 numbers within a single loop

```
n = (int)(15*RND) + 15*i - 14
```

This equation will generate numbers on [1,15], [16,30], [31,45], [46,60] and [61,75] as i varies from 1 to 5.

After the card has been randomly generated, the game is simulated by using a one dimensional array game[] of length 75. The array is initialized to 0's and then numbers are randomly generated between 1 and 75. The corresponding array element is replaced with a 1 and a variable gamect is increased by 1. A comparison is then made between corresponding elements of card[] and game[] to see if the number is on the card. If so, a variable cardct is increased by 1. The game is over when cardct reaches 24 and at that time, gamect contains the amount of numbers called.

The game was simulated 500,000 times and the results are printed in table 7. While it is theoretically possible to BINGO in 24 numbers, it is highly unlikely. In fact, the table shows that there never was a BINGO in less than 55 numbers and 32 percent of the time the game went to the very last number. It would appear that the cruise lines have a very good thing going for themselves since they start at 44 numbers and go up to at most 50 on a seven day cruise. At that point, a new group of passengers arrives and the game reverts back to the starting point of 44 numbers.

Let us now approach the problem from a purely theoretical standpoint.

Let

A = BINGO in exactly n numbers

Then

$P(A) = P(B1)P(B2 \mid B1)$

where

B1 = no BINGO in (n - 1) numbers

and

B2 = BINGO on the nth number

The P(B1) can be determined in the following manner. If one is going to BINGO on the nth number, 23 of the 24 numbers must be covered by the (n - 1)st number. This probability may be determined using the multiplication principle.

step 1 - select 23 of the 24 numbers - ${}_{24}C_{23} = 24$ ways

step 2 - select the other (n - 1) - 23 numbers - ${}_{51}C_{n-24}$ ways

$$P(B1) = \frac{(24)\, {}_{51}C_{n-24}}{{}_{75}C_{n-1}}$$

while

$$P(B2) = \frac{1}{76 - n}$$

since there are 76 - n numbers left at this point.

Consequently

$$P(A) = \frac{{}_{51}C_{n-24}}{{}_{75}C_{n-1}} * \frac{24}{76 - n}$$

Table 8 shows the above probability for values of n from 24 to 75.

A comparison of tables (7) and (8) indicates that the simulated results are very close to the theoretical results. In particular, the probability of a BINGO in 50 or less numbers is seen to be 0.0000047151. Keep in mind, however, that this is just for one individual person. Suppose n people are playing the game. Now what is the probability of a BINGO in 50 numbers or less?

To answer this question, let us find the probability of no winner in 50 numbers or less if n people are playing the game. Each individual may be considered to be a binomial experiment, either he wins or does not win. The n individuals then constitute repeating the binomial experiment n times. Consequently, the probability of exactly x successes is given by

$$P(x) = {}_nC_x \, p^x q^{n-x} \quad (1)$$

where success is no BINGO, p is the probability of success and q is the probability of failure. From table (8) we see that

$$p = .999995285 \quad \text{and} \quad q = .0000047151$$

Thus (1) becomes

$$P(x) = {}_nC_x \, (.999995285)x \, (.0000047151)^{n-x}$$

In the case where $x = n$, this becomes

$$P(n) = {}_nC_n \, (.999995285)n \, (.0000047151)^0$$

or $P(n) = (.999995285)^n$

1 - P(n) then gives the probability of at least 1 winner if n people are playing the game. Table (9) shows the results for selected values of n. As one can see, if 300 people are playing the game, a winner will occur in only 14 out of 10,000 games and even if 1000 are playing, there will only be a winner about 47 times in 10,000 games.

As mentioned earlier, the cruise lines appear to have a good thing going for themselves. Assuming 6 games per week, it will take approximately 1666 weeks or 32 years to play 10,000 games and since most cruises last a week, that's 1666 cruises with only 46 winners given 1000 players which, for a cruise, is definitely high.

Simulated Values Number of Games = 500000 Table 7

Nums Called	Bingos	Probability	Cumulative Probability
24	0	0.00000	0.00000
25	0	0.00000	0.00000
26	0	0.00000	0.00000
27	0	0.00000	0.00000
28	0	0.00000	0.00000
29	0	0.00000	0.00000
30	0	0.00000	0.00000
31	0	0.00000	0.00000
32	0	0.00000	0.00000
33	0	0.00000	0.00000
34	0	0.00000	0.00000
35	0	0.00000	0.00000
36	0	0.00000	0.00000
37	0	0.00000	0.00000
38	0	0.00000	0.00000
39	0	0.00000	0.00000
40	0	0.00000	0.00000
41	0	0.00000	0.00000
42	0	0.00000	0.00000
43	0	0.00000	0.00000
44	0	0.00000	0.00000
45	0	0.00000	0.00000
46	0	0.00000	0.00000
47	0	0.00000	0.00000
48	0	0.00000	0.00000
49	0	0.00000	0.00000
50	0	0.00000	0.00000
51	0	0.00000	0.00000
52	0	0.00000	0.00000
53	0	0.00000	0.00000
54	0	0.00000	0.00000
55	1	0.00000	0.00000
56	63	0.00013	0.00013
57	64	0.00013	0.00026
58	189	0.00038	0.00063
59	189	0.00038	0.00101
60	195	0.00039	0.00140
61	443	0.00089	0.00229
62	884	0.00177	0.00406
63	1012	0.00202	0.00608
64	1636	0.00327	0.00935
65	2288	0.00458	0.01393
66	3864	0.00773	0.02166
67	6002	0.01200	0.03366
68	10749	0.02150	0.05516
69	16745	0.03349	0.08865
70	21971	0.04394	0.13259
71	35104	0.07021	0.20280
72	52401	0.10480	0.30760
73	76081	0.15216	0.45976
74	110026	0.22005	0.67981
75	160093	0.32019	1.00000

Theoretical Values Table 8

Nums Called	Probability of BINGO in n numbers	Probability of BINGO in n numbers or less
24	0.0000000000	0.0000000000
25	0.0000000000	0.0000000000
26	0.0000000000	0.0000000000
27	0.0000000000	0.0000000000
28	0.0000000000	0.0000000000
29	0.0000000000	0.0000000000
30	0.0000000000	0.0000000000
31	0.0000000000	0.0000000000
32	0.0000000000	0.0000000000
33	0.0000000000	0.0000000000
34	0.0000000000	0.0000000000
35	0.0000000000	0.0000000000
36	0.0000000000	0.0000000000
37	0.0000000001	0.0000000001
38	0.0000000002	0.0000000004
39	0.0000000006	0.0000000010
40	0.0000000015	0.0000000024
41	0.0000000034	0.0000000059
42	0.0000000078	0.0000000137
43	0.0000000173	0.0000000311
44	0.0000000373	0.0000000683
45	0.0000000781	0.0000001464
46	0.0000001597	0.0000003061
47	0.0000003194	0.0000006255
48	0.0000006255	0.0000012509
49	0.0000012009	0.0000024518
50	0.0000022632	0.0000047151
51	0.0000041912	0.0000089063
52	0.0000076339	0.0000165402
53	0.0000136884	0.0000302287
54	0.0000241829	0.0000544116
55	0.0000421251	0.0000965367
56	0.0000724025	0.0001689392
57	0.0001228648	0.0002918040
58	0.0002059793	0.0004977833
59	0.0003413371	0.0008391204
60	0.0005594136	0.0013985340
61	0.0009071572	0.0023056912
62	0.0014562260	0.0037619172
63	0.0023150259	0.0060769431
64	0.0036461659	0.0097231090
65	0.0056915760	0.0154146850
66	0.0088083914	0.0242230764
67	0.0135198566	0.0377429330
68	0.0205870544	0.0583299873
69	0.0311093266	0.0894393139
70	0.0466639899	0.1361033038
71	0.0694995594	0.2056028631
72	0.1028014316	0.3084042947
73	0.1510551648	0.4594594595
74	0.2205405405	0.6800000000
75	0.3200000000	1.0000000000

Number Playing Bingo	Probability of no Winners	Prob Of at least 1 Winner
50	0.99976459	0.00235410
100	0.99952923	0.00047077
150	0.99929393	0.00070607
200	0.99905869	0.00094131
250	0.99882350	0.00117650
300	0.99858836	0.00141164
350	0.99835329	0.00164671
400	0.99811826	0.00188174
450	0.99788329	0.00211671
500	0.99764838	0.00235162
550	0.99741352	0.00258648
600	0.99717872	0.00282128
650	0.99694397	0.00305603
700	0.99670928	0.00329072
750	0.99647465	0.00352535
800	0.99624006	0.00375994
850	0.99600554	0.00399446
900	0.99577107	0.00422893
950	0.99553665	0.00446335
1000	0.99530299	0.00469771

Table 9

In conclusion, I would like to point out that the above examples are but a small sample of the wealth of material available to the instructor. Just take a look around you. Undoubtedly you will see a problem that can be solved using simulation and in all probability it will make an excellent programming project for beginners and experts alike. Happy Simulating.

Program Listings

```
*****************************************************************
/* random.h  -  Random Number Generator*/
#define MAX 2147483647L
float rnd(long int *seed)
{
 long int q, r, a, hi, lo, test;
  a = 16807L;
 q = MAX/a;
 r = MAX % a;
  hi = *seed/q;
 lo = *seed % q;
 test = a*lo - r*hi;
 if(test > 0)
 *seed = test;
else
```

```
*seed = test + MAX;
return((float)*seed/(float)MAX);
}
*****************************************************************
/* randint.h  -  Random Integer Generator
   This function will generate a random integer in
   the range [lo,hi] */

int randint(int lo,int hi,float seed)
{
 return((int)((hi - lo + 1)*seed) + lo);
}
*****************************************************************
/* contest.c - Keys in a Jar
  This simulation verifies that in a contest where 100
  people choose keys from a jar to win a new car, it
  makes no difference when you choose your key.  Each
  individual has a 1/100 chance of selecting the winning
  key */

#include<stdio.h>
#include"b:random.h"
#include<time.h>
void main()
{
 int winkey, winner, key, pos;
 int picks[101];
 int scores[101];
 float rnd();
 long int numsims, i;
 int j, count;
 long int seed, initseed;
 FILE *ptr;
 clock_t start,end;
 ptr = fopen("b:contest.dat","w");
 for(j=0;j<=100;j++)
 scores[j] = 0;
printf("Enter the seed: ");
scanf("%ld",&seed);
initseed = seed;
printf("Enter number of simulations: ");
scanf("%ld",&numsims);
start = clock();
for(i=1;i<=numsims;i++)
{
  for(j=1;j<=100;j++)
 picks[j] = 0;
 winkey = (int)(100*rnd(&seed)) + 1;
 winner = 0;
 pos = 1;
 while(!winner)
 {
```

```
 key = (int)(100*rnd(&seed)) + 1;
  count = 1;
  while(count < pos)
  {
      if(picks[count] == key)
      {
        count = 1;
        key = (int)(100*rnd(&seed)) + 1;
      }
      else
        count++;
  }
  picks[pos++] = key;
  if(key == winkey)
  {
      winner = 1;
      scores[pos-1]++;
  }
  } }
  end = clock();
  printf("Seed = %ld        Number of Simulations = %ld\n",
  initseed,numsims);
  fprintf(ptr,"Seed = %ld        Number of Simulations = %ld\n",
  initseed,numsims);
  printf("Time of simulation = %f seconds\n\n",(end-start)/CLK_TCK);
  fprintf(ptr,"Time of simulation = %f seconds\n\n",(end-start)/CLK_TCK);
  for(j = 1;j<=20;j++)
 {
printf("%3d %5d  %3d %5d  %3d %5d  %3d %5d  %3d %5d\n",
 j,scores[j],j+20,scores[j+20],j+40,scores[j+40],
 j+60,scores[j+60],j+80,scores[j+80]);
fprintf(ptr,"%3d %5d  %3d %5d  %3d %5d  %3d %5d  %3d %5d\n"
 ,j,scores[j],j+20,scores[j+20],j+40,scores[j+40],
 j+60,scores[j+60],j+80,scores[j+80]);
 }
  fclose(ptr);
}
********************************************************************* *****
/* dice.c  Dizzy Dice
A simulation to determine which is more likely, rolling at
least 1 six in 4 rolls of a single die or rolling at least
1 twelve in 24 rolls of a pair of dice. */

#include<stdio.h>
#include<conio.h>
#include"b:random.h"
#include"b:randint.h"

void main()
{
 long int seed, nosims, i,c1,c2;
 int   d[5],d1[25],d2[25],j,flag;
 float ave1,ave2;
  FILE *ptr;
  ptr = fopen("b:dice.dat","w");
  printf("Enter a seed for the random number generator ");
  scanf("%ld",&seed);
  printf("Seed = %ld\n",seed);
  fprintf(ptr,"Seed = %ld\n\n\n",seed);
  printf("Enter the number of simulations ");
  scanf("%ld",&nosims);
  fprintf(ptr,"Number of simulations = %ld\n\n",nosims);
 c1 = 0;
 c2 = 0;
 for(i=1;i<=nosims;i++)
  { for(j=1;j<=4;j++)
  {
 d[j] = randint(1,6,rnd(&seed));
 }
 flag = 0;
 for(j=1;j<=4;j++)
 {
 if(d[j] == 6)
     flag = 1;
 }
 if(flag)
 c1++;
 for(j=1;j<=24;j++)
 {
 d1[j] = randint(1,6,rnd(&seed));
 d2[j] = randint(1,6,rnd(&seed));
 }
 flag = 0;
 for(j=1;j<=24;j++)
 {
 if(d1[j]+d2[j] == 12)
     flag = 1;
 }
 if(flag)
 c2++;
 }
 ave1 = (float)c1/(float)nosims;
 ave2 = (float)c2/(float)nosims;
 printf("At least 1 six occurred in 4 rolls of a single die %8.6f\n",ave1);
 printf("percent of the time\n\n");
  fprintf(ptr,"At least 1 six occurred in 4 rolls of a single die %8.6f\n",
 ave1);
 fprintf(ptr,"percent of the time\n\n");
 printf("At least 1 twelve occurred in 24 rolls of a pair of dice \n");
 printf("%8.6f percent of the time\n\n",ave2);
 fprintf(ptr,"At least 1 twelve occurred in 24 rolls of a pair of dice \n");
 fprintf(ptr,"%8.6f percent of the time\n\n",ave2);
 fclose(ptr);
}
```

BEST COPY AVAILABLE

```
****************************************************************** ****
/* aces.c  Chase the Ace
A simulation to determine the average number of cards
that must be turned over before encountering an ace. */


#include<stdio.h>
#include<conio.h>
#include"b:random.h"
#include"b:randint.h"

void main()
{
 long int seed, nosims, i, expected;
 int deck[53], j, m, temp, c;
 float ave;
 char ch;
 FILE *ptr;
 ch = 'y';
 ptr = fopen("b:aces.dat","w");
 for(j=1;j<53;j++)
   deck[j] = j;
printf("Enter a seed for the random number generator ");
scanf("%ld",&seed);
printf("Seed = %ld\n",seed);
fprintf(ptr,"Seed = %ld\n\n\n",seed);
while(ch == 'y')
{
 printf("Enter the number of simulations ");
 scanf("%ld",&nosims);
 expected = 0;
 for(i=1;i<=nosims;i++)
 {
 for(j=1;j<53;j++)
 {
      m = randint(1,52,rnd(&seed));
      temp = deck[j];
      deck[j] = deck[m];
      deck[m] = temp;
 }
 c = 1;
  while(deck[c]%13 != 1) c++;
   expected += c;
 }
 ave = (float)expected/(float)nosims;
 printf("Number of simulations = %ld\n",nosims);
 fprintf(ptr,"Number of simulations = %ld\n",nosims);
 printf("The expected number of cards is %9.6f\n",ave);
 fprintf(ptr,"The expected number of cards is %9.6f\n\n\n",ave);
 printf("Would you like another simulation? y or n?\n ");
 ch = getch();
 }
 fclose(ptr);
}

***************************************************************** ****
/* hats.c  Hatcheck Problem */

/* If hats belonging to n men are randomly returned, how likely is it
that at least one will get his own hat back?*/


#include<stdio.h>
#include"b:random.h"
#include"b:randint.h"

void main()
{
 int hat[53],match,j,k,m,temp;
 long int nosims,i,seed,c;
 FILE *ptr;
 ptr = fopen("b:hats.dat","w");
 printf("Enter the number of simulations ");
 scanf("%ld",&nosims);
 printf("Enter a seed for the random number generator ");
 scanf("%ld",&seed);
 fprintf(ptr,"Seed = %ld\n",seed);
 for(j=2;j<=52;j++)
 {
 for(k=1;k<=j;k++)
 hat[k] = k;
 c = 0;
 for(i=1;i<=nosims;i++)
 {
 for(k=1;k<=j;k++)
 {
     m = randint(1,j,rnd(&seed));
     temp = hat[k];
     hat[k] = hat[m];
     hat[m] = temp;
 }
  match = 0;
  for(k=1;k<=j;k++)
   if(hat[k] == k)
      match = 1;
   if(match)
   c++;
 }
 printf("No men = %2d    Percent of matches = %f\n",j,(float)c/(float)nosims);
 fprintf(ptr,"No men = %2d    Percent of matches = %f\n",j,
 (float)c/(float)nosims);
 }
 fclose(ptr);
}
```

```
****************************************************************** ****
/* birthday.c  Birthday Problem
program to find the probability that in a room containing
n people, at least two have the same birthday. */

#include<stdio.h>
#include"b:random.h"
#include"b:randint.h"
void main()
{
 int c;
 int i,j;
 int b[366];
 int flag;
 long int seed,n;
 int r1;
 int k,m;
 float rnd(),percent;
 FILE *ptr;
 ptr = fopen("b:birthday.dat","w");
 printf("Enter a seed for the random number generator ");
 scanf("%ld",&seed);
 fprintf(ptr,"Seed = %ld\n",seed);
 printf("Enter the number of simulations desired ");
 scanf("%ld",&n);
 fprintf(ptr,"No. of simulations = %ld\n",n);
 for(k=2;k<51;k++)
 {
 c = 0;
 for(i=0;i<n;i++)
 {
 for(j=1;j<=k;j++)
      b[j] = randint(1,365,rnd(&seed));
 flag = 0;
 for(j=1;j<k;j++)
 {
      for(m=j+1;m<=k;m++)
       if(b[j] == b[m])
       flag = 1;
 }
 if (flag == 1)
      c++;
 }
 percent = (float)c/n;
 printf("If a room contains %d people, the probability\n",k);
 printf("that at least two have the same birthday is %8.6f\n\n",percent);
 fprintf(ptr,"If a room contains %d people, the probability\n",k);
 fprintf(ptr,"that at least two have the same birthday is %8.6f\n\n",percent);
 }
 fclose(ptr);
}
****************************************************************** ****
```

```
/* pete.c  St. Petersburg Paradox
A simulation to determine the average winnings in the St
Petersburg Paradox. */


#include<stdio.h>
#include<conio.h>
#include"b:random.h"
#include"b:randint.h"

void main()
{
 long int seed, nosims, i, sum,win;
 int   m;
 float ave;
 char ch;
 FILE *ptr;
 ch = 'y';
 ptr = fopen("b:pete.dat","w");
 printf("Enter a seed for the random number generator ");
 scanf("%ld",&seed);
 printf("Seed = %ld\n",seed);
 fprintf(ptr,"Seed = %ld\n\n\n",seed);
 while(ch == 'y')  {
 printf("Enter the number of simulations ");
 scanf("%ld",&nosims);
 sum = 0;
 for(i=1;i<=nosims;i++)
 {
 m = 1;
 win = 1;
 while (m)
 {
      m = randint(0,1,rnd(&seed));
      win *= 2;
 }
 sum += win;
 }
 ave = (float)sum/(float)nosims;
  printf("Number of simulations = %ld\n",nosims);
 fprintf(ptr,"Number of simulations = %ld\n",nosims);
 printf("The average winnings is  %9.2f\n\n\n",ave);
 fprintf(ptr,"The average winnings is %9.2f\n\n\n",ave);
 printf("Would you like another simulation? y or n?\n ");
 ch = getch();
 }
 fclose(ptr);
}

****************************************************************** ****
/* Program to simulate Cover-all BINGO */
```

```
#include<stdio.h>

long int seed;
int freq[76],card[76],game[76];
float percent[76],relfreq[76];
FILE *fptr;

void main()
{
 int count,number,i,l,fill_card(),play_game();print_results();
 fptr = fopen("bingo.txt","w");
 printf("Input seed for random number generator.\n");
 printf("The seed should be odd and less than 2147483647.\n");
 scanf("%d",&seed);
 printf("SEED = %d\n",seed);
 fprintf(fptr,"SEED + %d\n",seed);
 printf("Input the number of BINGO games to be played.\n");
 scanf("%d",&number);
 fprintf(fptr,"NUMBER OF GAMES = %d\n\n",number)_;
 printf("NUMBER OF GAMES = %d\n\n",number);
 for(i=1;i<= 75;i++)
 freq[i] = 0;
 fillcard();
 for(l=1;l<= number;l++)
 {
  for(i=1;i<= 75;i++)
  {
  game[i] = 0;
  }
  count = play_game();
  freq[count] = freq[count] + 1;
  }
  for(i=1;i<=75;i++)
  percent[i] = (float)(freq[i]/(float)number;
 relfreq[i] = 0;
 for(i=2;i<=75;i++)
  relfreq[i] = elfreq[i-1] + percent[i];
 print_results();
 }
**********************************************************************

/* Function to Randomly Generate a BINGO Card */

fill_card()
{
int i,j,n,flag;
float random();
for(i=1;i<=75;i++)
{ for(j=1;j<=5;j++)
 {
 if((i != 3) | | (j != 1)
 {
 flag = 0;
 while(flag == 0)
 {
 n = (int)(15*random(&seed)) + 15*i - 14;
 if(card[n] == 0)
 {
 card[n] = 1;
 flag = 1;
 } /* end of 2nd if */
 } /* end of while */
 } /* end of 1st if */
 } /* end of for j */
 } /* end of for i */
} /* end of function */

****************************************************************** ****

/* Function to Play the Game of BINGO */

play_game()
{
 int cardct,gamect,flag,m;
 float random();
 cardct = 0;
 gamect = 0;
 while(cardct < 24)
 {
 flag = 0;
 while(flag == 0)
 {
 m = (int)(random(&seed) * 75) + 1
 if(game[m] == 0)
 {
 game[m] = 1;
 gamect = gamect + 1;
 flag = 1;
 if(card[m] == 1)
 cardct = cardct + 1;
 } /* end of if */
 } /* end of 2nd while */
  } /* end of 1st while */
 return(gamect();
} /* end of function */
****************************************************************** *****

print_results()
{
 int i;
 for(i=24;i<=75;i++)
 {
 printf("%2d    %6d    %7.5f    %7.5f\n",i,freq[i],percent[i],
       relfreq[i]);
  fprintf(fptr,"%2d    %6d    %7.5f    %7.5f\n",i,freq[i],
  percent[i],relfreq[i]);
  } /* end of for */
  fclose(fptr);
} /* end of function */
** ********************************************************************** ***
/* comb.c
This program calculates the theoretical probability
of getting a cover-all BINGO in exactly n numbers as
well as the probability of BINGO in n or less numbers.*/

#include<stdio.h>
void main()
{
  FILE *fptr;
  double prob[76],rel[76];
  double comb();
  int i;

  rel[23] = 0;
  fptr = fopen("a:prob.txt","w");
  for(i = 24;i <=75;i++)
  {
  prob[i] = 24 * comb(51,i - 24) / comb(75,i-1) / (76 - i);
  rel[i] = rel[i-1] + prob[i];
  }
  for(i = 24;i <=75;i++)
  {
  printf("%3d %15.10f %15.10f\n",i,prob[i],rel[i]);
  fprintf(fptr,"%3d %15.10f %15.10f\n",i,prob[i],rel[i]); }
  fclose(fptr);
}
********************************************************************** ****
/* This function calculates the number of combinations possible when
r objects are selected from n objects */

double comb(n,r)
 int n,r;
{
  int i,max;
  double x;
  if((n - r) > r)
  max = r;
else
  max = (n - r);
x = 1.0;
for(i = 1;i <= max;i++)
  x = x * (n + 1 - i) / i;
return(x);
}
```

```
*************************************************************** *****
/* binom.c
This program calculates the probability of someone getting
a cover-all BINGO in 50 numbers or less if n people are
playing the games.*/

#include<stdio.h>
#include<math.h>
void main()
{
 FILE *fptr;
 double p,prob,pow();
 int i,n;
 fptr = fopen("a:binom.txt","w");
 printf("Input the probability of one person winning");
 scanf("%f",&prob);
 fprintf(fptr,"            n               P(0)
 1 - P(0)\n");
 printf("            n               P(0)
 1 - P(0)\n");
 for (n = 50;n<=1000;n+=50)
 {
 p = pow(prob,(double)n);
 fprintf(fptr,"%6d %13.8f %15.8f\n ,n,p,1-p);
 printf("%6d %13.8f %15.8f\n",n,p,1-9);
 }
}
*************************************************************************** *****
```

References

1. S.K. Park and K. Miller, Random Number Generators: Good ones are hard to find, Communications of the ACM 31 (1988) 1192-1201.

2. Kemeny, Snell & Thompson, Introduction to Finite Mathematics, Prentice Hall Publishing Co.

Moving Small Colleges into the Information Age: Problems, Resources, and Strategies for Success

Dian Thompson
Computing Center Director
Edgar H. Thompson
Director of Elementary Education
Emory & Henry College
Emory, VA 24327
703–944–4121
John Stuckey
Director of University Computing
Washington and Lee University
Lexington VA 24450
703–463–8841
jstuckey@wlu.edu

Introduction:

Coming from small colleges, we often feel that the push for technology at universities originates from content areas where the use of technology is endemic to the structure of their disciplines and research, e.g., engineering, the sciences, mathematics. Once resources are acquired and used by these disciplines, it appears that there is a spread of effect to less technologically oriented disciplines, who then begin to make adaptations in order to use the new resources. At small colleges like ours, where the focus is much more on teaching than research, the push for more technology also tends to come from the sciences, but because of smaller student numbers, the kind of technology used and requested is often idiosyncratic, with only very narrow applications, and then often relevant only to one researcher's endeavors, as opposed to broader faculty and student use.

In our opinion the difference in the focus of small colleges creates a very real problem when it comes to technology. To remain competitive educationally in terms of what is being offered to students, small colleges must plug into technology in ways they often haven't done before. Since money for technology doesn't come along as a research spin–off as it may at the university level and since grants generally don't allow for substantial purchases of hardware and software, the funding for technological change in small colleges has to come from the operating budget of the institution. A conscious, strategic decision by the college must be made to allocate these funds—funds that are usually already stretched. Considering that many administrative officers at small colleges do not come from disciplines where technology plays a major role, though they may be sympathetic to such needs, they may not understand these needs from their own personal or professional experience. Further, since there may be little push for new technology from faculty,an administrative officer could easily conclude that there is no great need, and that limited funds might be best spent in other ways.

Because computer resources and technology aren't always a matter of strategic planning, little thought may be put into hardware and software purchases which may be made with the attitude: "We've bought something new, and we won't need to bother with this again for years." Additionally, new purchases may have very limited applications if bought without reference to an overall plan. We know of one college that has 40 reports produced for the registrar's office on aging administrative software. Though current PC spreadsheet software would make this report much easier to read, the administrators have gotten used to reading a convoluted report and see no particular need to change.

Another problem we think exists at small colleges is the use of software that is not "friendly." This may be due to the use of antiquated versions of software or the purchase of a lesser product for a reduced cost. Since the overall focus of small colleges is not usually towards technology, staff may not be able to do important office functions on the computer because they don't do it often enough to learn how to manipulate the software. For example, one of our offices needed to generate 200 letters. They used a memory typewriter and added each of the 200 addresses, one at a time, because they didn't have the time to learn how to use a merge procedure on the PC or use the college's administrative system. As a result of behaviors and time limitations such as these, the full potential of a lot of software is lost.

Emory & Henry is currently wrestling with these technologically–oriented issues. Our president, Dr. Tom Morris, knows what kinds of things need to be done. He has already allocated more money for both hardware and software. Further, we are waiting to hear if we have received a grant that will affect the way we allocate funds for hardware, which in turn will affect the way we buy software, which in turn will affect the overall strategic decision that is going to have to be made about where we are going in the future. While we are waiting for these hardware and software purchases to be approved, our administration is getting a lot of conflicting information. Different departments have their own needs and agendas, which may not necessarily jive with the overall needs of the college. For instance, one department at Emory & Henry wants to network everyone. This is fine, but how can a small college contemplate such networking when the basic administrative and instructional needs for computer technology haven't yet been met? Also, we are beginning to think that being able to use top notch technology may ultimately become something available only at universities, after you have acquired your basic, liberal arts education at a small college. If this scenario becomes a reality, then it will affect how computers and technology are integrated into the educational programs and the administrative offices of small colleges.

We believe that many small, rural colleges are also dealing with some of the same issues and conflicts that we are. To find out just exactly where some of these institutions stand on these issues, we asked several small colleges to answer a series of questions (see Findings).

Procedures:

For our survey, we decided to look at small, independent colleges in our geographic region. We felt this was important because the resources of the region were similar (coal, light industry, and/or farming) and, possibly, the funding sources of the colleges were similar. Most of the colleges in our region were in rural or small city settings similar to Emory & Henry. We also felt backgrounds of students, faculty, and staff might be similar. Six of the colleges belong to the Appalachian College Program, an organization originating at the University of Kentucky which is designed to promote faculty growth and college assessment in the Appalachian region. Three colleges were in southern Virginia and not part of the Appalachian College Program, but were similar in scope with Emory & Henry. We discussed student enrollment as a limiting factor, but we decided it was not critical to the outcome of our survey. In fact, we were interested in colleges with a slightly larger enrollment because we wanted to see if they had any advantages that a smaller school like Emory & Henry didn't have.

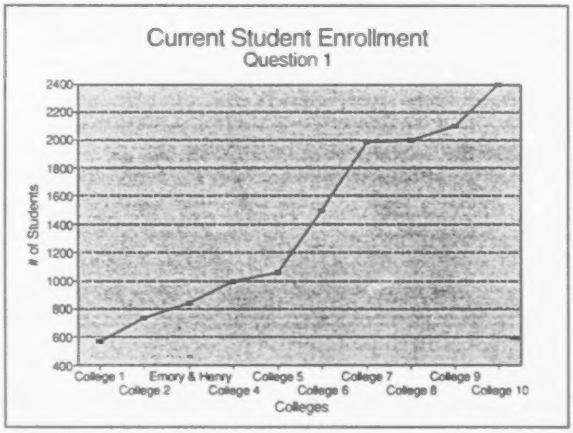
We decided that the best and quickest way to get the information for our survey was to fax the questions to a computing resource person at each college and then call them at a predetermined time to talk over the questions. This insured that the questions would be answered and that we would get a high response rate. We realized that the results would be subject to the interviewer's interpretations of the phone conversations; however, it seemed more likely that the results would be more accurate than a written survey, because the interviewer could ask questions if the response was vague or unclear.

Interviews were conducted from March 31, 1994 through April 6, 1994. Two colleges requested that their interview times be rescheduled. One college's contact person could never be reached. The interview times ranged from 30 minutes to one hour for the nine colleges interviewed. Handwritten notes were taken by an interviewer at the time of the interview.

Findings:

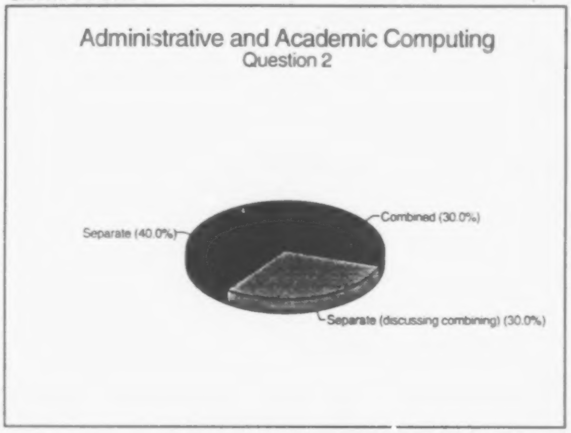
Following is a composite summary of the answers to each of the questions asked on our survey. Emory & Henry's status in relation to these questions is included separately because one of our goals in this paper is to determine how our computing resources can be most efficiently evaluated.

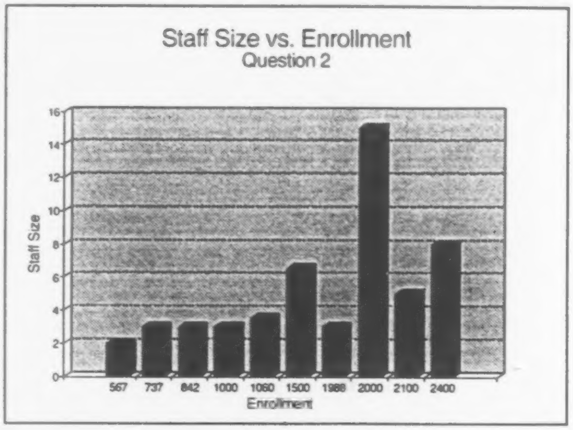
1. What is your current student enrollment?



Reported full–time enrollment at the colleges ranged from 567 to 2400 students. Four of the colleges had an enrollment of approximately 1000 or fewer. Two colleges with a larger enrollment number had a graduate program. Emory & Henry has 842 students.

2. Does your college separate administrative and academic computing into two distinct areas?





All but two colleges officially break their administrative and academic computing into two separate entities. Of the two colleges that don't separate the computing resources, one college usually divides staff members' job responsibilities into academic or administrative. The other college feels that academic computing suffers because the emphasis is usually on administrative computing.

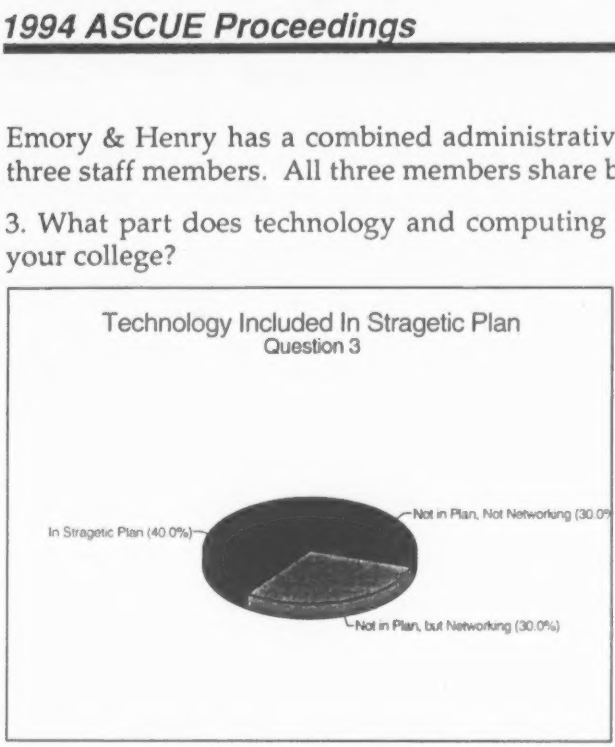
The colleges that separate their computing resources all feel it would be beneficial to have contact with each other and share resources. One college has decided to combine administrative and academic computing so resources and staff expertise can be shared. Two other colleges are discussing the possibility of combining administrative and academic computing as well, because of duplication of staff expertise and resources.

Three colleges have no immediate plans to combine resources even though computing staff in two of these colleges would like contact between the two areas. One college is content to keep administrative and academic separate, because they already share resources with each other.

Staff sizes for combined administrative and academic areas range from two people to 15 people. With the exception of one college, larger colleges typically have larger staff sizes. Academic staff sizes range from one to five people and administrative staff sizes range from zero to 10 people (the college with no administrative staff has a contract with the administrative software company for maintenance and help). One college with an enrollment of approximately 1000 has two faculty members that take care of academic computing.

Emory & Henry has a combined administrative and academiccomputing center with three staff members. All three members share both academic and administrative jobs.

3. What part does technology and computing resources play in the strategic plan of your college?



Five colleges do not mention technology and computing resources in their strategic plans; however, three of these colleges have made commitments to network their campuses. The administrators of these three colleges are aware of the need for current technology on their campuses and have appropriated the money for this to happen. One of the five colleges currently has a committee underway to include technology in the strategic plan. Another college has a new president arriving next year and is unsure of his feelings on technological advancements. In the past, this college has budgeted for campus networking, but it was not a part of the strategic plan.

Four colleges, all with larger enrollments, do have computing resources and technology included in the strategic plan and are involved in or have finished networking their campuses. In one of them, technology is the primary goal of their strategic plan, and another has incorporated technology for the last 15 years.

Emory & Henry's strategic plan currently does not account for information technology. We are about to undergo a self–study and plan to discuss implementation of technology during that time.

The colleges have varying degrees of demand for improved computing and technology. Most prevalent, is the demand for access to E–mail and/or the Internet. Many new faculty and staff to the college who come to the college from other places where more extensive technological capabilities were available arrive with high expectations and needs. Several colleges reported a desire for technology upgrades as teaching or office tools.

Four colleges have a campus–wide demand for improving and upgrading current technology. One of these colleges even has a faculty committee to choose and upgrade hardware and software for the campus.

Three colleges have most members on campus interested in improving what they currently have, but on one of these college campuses, there are still some faculty who are indifferent. Even though everyone may not have a PC at their desk, there are still many who want the capability of using a PC.

4. How widespread is the demand for improving your computers and technology—both academic and administrative—at your institution? What departments are pushing hard? Which ones are indifferent?

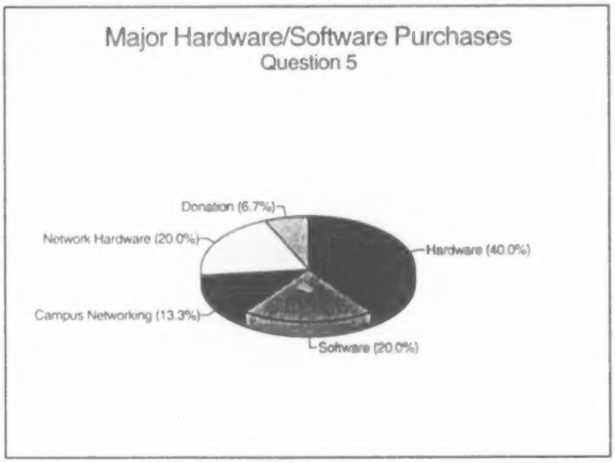
✓ = Wants Improvement | ♥ = Pushing for Change | ⌂ = Indifferent

	Colleges									
	1	2	E&H	4	5	6	7	8	9	10
Improvement	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Internet	✓				✓	✓		✓		✓
E-mail	✓				✓	✓	✓	✓	✓	✓
Depts:										
Campus-wide						♥		♥	♥	♥
Business	♥		♥				♥			
Sciences	♥		♥		♥		♥		♥	
English										♥
Psychology										♥
Computer Science					♥		♥			
Math			♥		♥					
Music			♥				⌂		♥	
History							⌂			
Admin:										
Admissions	♥	♥	♥							
Development		♥								
Library					♥					
Financial Aid	♥		♥							

Two colleges have faculty members who want improved technology, but probablyan equal number who are indifferent. One college which already has its campus networked falls in this category, and the contact person believes the lack of demand for improved technology is probably due to lack of training of what's already available.

Five of the colleges felt the demand from all their departments was great to improve computing resources. At least one to two people in every department wanted to add or change what was currently available. The biggest demand for change from all the colleges was in the academic business and science departments and the admissions office. Only one college had two departments (music and history) which were indifferent to additional computing.

Most of the academic departments at Emory & Henry, particularly our science, business, math, and music departments, are asking for increased computing resources. Most of our administrative offices are frustrated with old hardware and software and would welcome new technology.



5. Have you recently made a major hardware or software purchase? Are you happy with that purchase and why?

Two colleges had recently bought equipment for campus-wide networking. This usually included fiber optic cable as a backbone and other types of wiring for individual connections within buildings. Servers, routers, and associated electronics were included in the purchases.

Some colleges had made major hardware purchases—two had bought administrative systems, one had bought a 486 lab, one a VAX for student use, and one an administrative LAN. Four colleges already had networks in place and three had bought hardware to enhance the existing network.

One college had no major computer purchase, but did have a donation of 386 PCs and a laser printer.

Every college was happy with purchases they made even though most had problems with installation. The benefits brought to the campus, however, outweighed the trouble in setting up the new purchases. One college contracted through a local vendor to have their network installed, and the installation was extremely smooth. For them, it was worth the money invested to contract through an outside source.

Emory & Henry has recently purchased two hard disk drives and a tape drive for an IBM System/36. This was done to extend the usability of the current system for 12 to 18 months while the college tries to decide what administrative hardware to buy in the future.

Next year's purchases for six colleges will be to continue setting up their networks. For four colleges this means adding to the existing network, such as, wiring to offices, dorms, and labs. It also means for one college the purchasing of hardware to use on the network. One of the colleges included in this network category will not be purchasing equipment but will be paying off networking bills.

Of the three colleges not pursuing the expansion of networks, one.plans to make an administrative hardware purchase to enhance handling of transcripts. The other two colleges plan to upgrade their current resources with purchases suchas new PCs for labs, faculty, and/or staff.

Emory & Henry has applied for a grant to upgrade our student labs. We are also beginning to look at an administrative system upgrade.

6. What computing resources do you anticipate purchasing in the next year? In the next five years?

	One year	Three years	Five years
College 1		Automate library, upgrade administrative software	
College 2	Additional PCs, maybe multimedia	Evaluate administrative software	
Emory & Henry	Evaluate administrative software/hardware, grant pending for lab upgrade		
College 4	Upgrade academic lab, grant pending for faculty hardware		Whole campus networked
College 5	Four PCs per division		Whole campus networked
College 6	Fiber and electronics for network		Whole campus networked
College 7	Administrative hardware		Whole campus networked
College 8	Upgrade PCs and printer, networking in dorms		
College 9	Connect offices to network		
College 10	Add another academic lab, color printer, scanners, multimedia		Network link to city schools

Three years

Two colleges plan to re–evaluate their administrative software and hardware within the next three years and possibly buy a new system. One college hopes to automate their library within three years.

Five years

One college in the next five years hopes to link to the city's school system so the public schools and college can share resources. Three colleges hope to have their campus–wide networks in place (one of these colleges is not interested in wiring in the dorms). The rest of the colleges have no structured five–year plan. Most of them hope to have their campuses networked if they are currently involved in doing so.

At this time, Emory & Henry has no specific computer technology plans for the next five years. We are discussing networking our campus, however.

7. Is learning time ever factored into a hardware and/or software purchase? This learning can be in seminars or class or one–on–one tutorials or on your own.

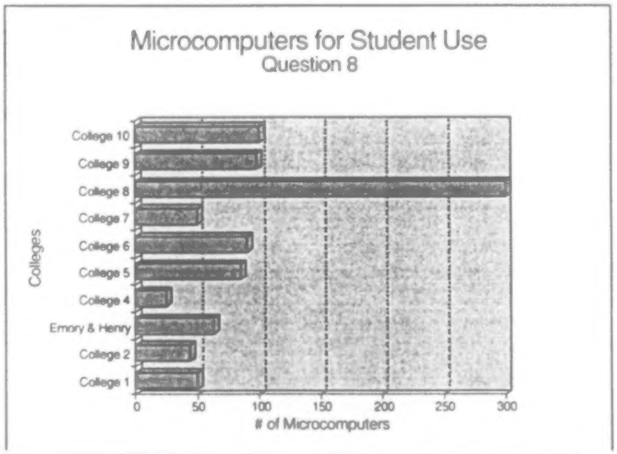
	Colleges									
	1	2	E&H	4	5	6	7	8	9	10
Need more training	✓	✓	✓	✓	✓	✓	✓		✓	✓
Adequate training								✓		
Specialized one-time workshops	✓	✓	✓		✓		✓	✓	✓	
On-going workshops				✓		✓	✓	✓		✓
Word processing workshops						✓	✓	✓		
Internet/E-mail workshops	✓					✓	✓	✓	✓	✓

Only two of the schools surveyed did not usually involve themselves in training activities. One of these schools did not because an administrator said it wasn't necessary. Of the schools who did provide training, only one felt comfortablewith the level of training they were doing. Most training is provided through workshops that are aimed at teaching particular software packages, either new purchases or upgrades to software already available. Some classes are provided for word processing. Also, special workshops are provided on special topics as needed. For example, one school provided several workshops when they attached to the Internet. Five of the schools expressed a need for more people designated to do training.

We believe that Emory & Henry needs to factor learning time into new hardware and software purchases. The computing center staff does this when possible, but usually, there isn't time.

8. How many microcomputers does your college make available for student use? Any idea what proportion of students own their own computers?

Total microcomputers for student use average between 50 and 100 per campus. One college has 300 microcomputers, four have 75 – 100, three have 40 – 75, and one has 25.



Only two colleges were sure of how many students owned their own microcomputers. One college had 80% of their students who had their own, and the other college had 30% of students with their own computers. Most contact persons didn't know for certain, but they knew that some students did bring their own computers. The feeling on all campuses, except for one, was that most students relied on the college for computing resources.

Emory & Henry has approximately 60 computers for student use. We do not know how many students have their own computers, but we do know that the number increases each year.

9. What's the campus slant on the choice between Macintosh and Intel–based micros?

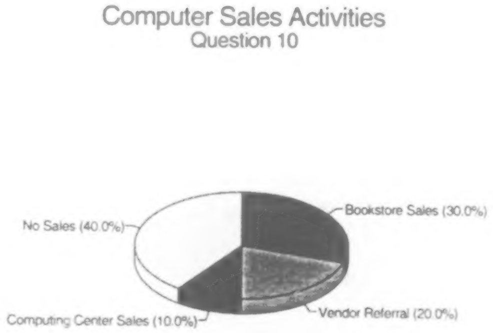
Lack of Macintosh presence due to:	Number
IBM has broader range of software	2
Purchases dependent upon individual department needs	2
Purchase and support of Apple products is bad	1
One more platform to maintain	2
Started with IBM products	3

All colleges had Intel–based systems as the most prominent type of microcomputer on campus. All colleges did, however, have some Macintosh presence on their campus, whether it was one or two in a student lab, a whole lab, or some in faculty or staff offices. Two colleges want more Macintoshes than they currently have. The reasons why the number of Macintoshes isn't greater on other campuses is that IBM offers a broader range of software (two colleges), purchases are dependent on how much a particular academic or administrative department knows about what software is available (two colleges), purchase and support of Apple products has been bad in comparison to IBM products (one college), it adds one more platform to maintain for the computing staff (two colleges), and IBM products are what they started with (two colleges).

Emory & Henry is almost exclusively Intel–based. Only one or two Apple machines are available for students on campus. The first hardware purchases were IBM machines, which have worked satisfactorily, so there has been no perceived need to use a Macintosh platform.

10. Does the college participate in any computer–sales activities, either to faculty/staff or students? If so, what are they?

Only one of the schools surveyed engaged in sales activities. This



school coordinated placing computer purchases for students through a local business. All of the rest of the schools did not engage in sales activities, though the bookstores at three of them sold some hardware and software. One school connected users with a local vendor, and another one worked as an intermediary between users and Apple Collegiate Partners III. Discounts were available for faculty and staff at this institution.

Emory & Henry does not have any computer–sales activities.

11. Networks are talked about a lot. Do you have any local–area networks? A campus backbone linking such networks? Connection to an outside network? What is local thinking and planning about networks?

	Colleges									
	1	**2**	**E&H**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Fiber backbone	✓	✓		✓		✓	✓		✓	✓
% of campus connected	70	40		50		50	50		100	66
Outside connection (Internet)	✓				✓	✓		✓		✓
E-mail only							✓		✓	

One college, with a smaller enrollment than most surveyed, had one campus–wide local–area network (LAN). One college had one lab with a LAN, two colleges had two LANs, two colleges had three LANs, and one college had eight LANs. Of the colleges any type of networking on campus, all but two have academic as well asadministrative computers on LANs. Two colleges currently have all or a portion of their administrative offices on a network but are not working towards an academic network.

Seven colleges have a fiber–optic backbone in place and have linked or are linking their networks together. Three colleges have at least half the campus linked to a fiber–optic backbone. One college is in the process of installing a backbone. One college has no backbone for networks. A college that has been installing networks for a number of years is wiring all the dorms and buildings first and then installing the backbone last, unlike the other colleges who are installing networks.

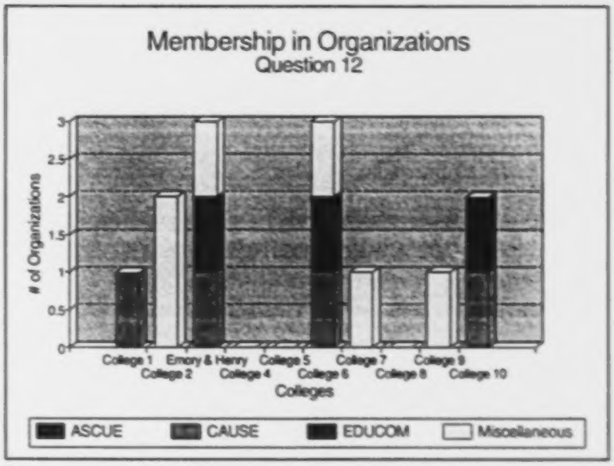
Five colleges have a connection to Internet as an outside network source. Two colleges have E–mail access on campus but no outside network source. Five colleges do not have Internet access.

All colleges that are installing networks plan to continue until they are completed. Only one college currently installing is not planning to install network capability in the dorms. One college that installed a campus–wide network in 1991 is about to re–evaluate what they've done to see if they want to do any additional installation. One college is finished installing the campus network and is not planning to do any upgrades other than add new hardware to the network.

Emory & Henry has one lab with a LAN and one administrative office with Macintoshes connected through LocalTalk. We have no campus backbone. We have limited Internet access through our library, but it is not used by the rest of the campus. We have been talking about networking, E–mail, and Internet access for offices and labs.

12. Do you belong to EDUCOM? Why or why not?

Of the schools surveyed, only two belonged to EDUCOM. These two liked EDUCOM but one felt that it was too technologically oriented, and the other one particularly liked CAUSE and ASCUE. Several of the others expressed an interest in knowing more about the organization. One didn't know what it was. Another one felt that EDUCOM's focus was too much on universities and not small colleges. This school had been to ASCUE.



Emory & Henry belongs to EDUCOM. Last summer we used their organization to set up an evaluation of our academic and administrative computing resources. We needed a direction in which to move, and this evaluation provided a starting point.

Recommendations:

The results of our survey suggest that small colleges want to make more productive use of computing resources just as we do at Emory & Henry. To doso, we think we need to do the following things:

1. Make computing resources a part of the strategic planning of the college.

Apparently, computing resources don't necessarily have to be included in a college's strategic plan, but there must be a wide–spread commitment by the faculty, the staff, or the administration. Probably at Emory & Henry, our best move is to incorporate technology into our strategic plan. In addition, colleges like ours need to decide whether to fund generic computer resources or leave such purchases to individual departments. Commit-

BEST COPY AVAILABLE

tees made up of various constituencies within the college and/or outside sources (local vendors) need to be recruited to evaluate software and hardware before purchase.

2. Colleges must realistically assess where they are if they are to have a base–line from which to extrapolate where they need to go.

A comparative assessment of current resources is important to establishing a baseline that is realistic. In addition, money needs to be budgeted to send college members to appropriate computer and technologically oriented conferences, e.g., ASCUE, CAUSE, EDUCOM, SHARE, IAT, etc. When these people see what is available, they will be in a better position to evaluate the needs of the college. This will lead to better decision making for the future.

3. Provide access to the outside world.

It appears from our survey that probably the single most important factor in whether a campus was expanding its technology was the access to E–mail and/or the Internet. Every campus but one that had this access had a high demand from users to increase and upgrade existing computer technology. The ability of a small college to communicate and connect with the outside world would make us feel less isolated as well as tremendously increasing our limited resources.

4. Make computer resource training a primary goal.

From our survey, learning time was considered a critical factor in how well received computer technology was on a campus. Colleges that could provide training for their faculty, students, and staff had a more extensively used system than colleges who let the users learn on their own. One college who has done a great deal of networking on campus and has had students buy their own PCs, felt learning time was a critical factor in having happy users. They know now that anything else they add will also have to have learning time calculated into the installation. Lack of training for those who must use new technology undermines all of our best efforts. There is no short cut. If colleges want new software and technologies to be incorporated into the daily routines of the institution, time and money must be budgeted for the appropriate training in how to use the new technologies and then enough time has to be allowed for supervised trial and error and practice until people become accustomed to the new things they are being asked to do. Just providing technology and ordering that it be used will not get the job done.

The research we have checked on staff development in educational settings provides us some direction. This research suggests that "without concrete, tangible guidance, innovative ideas typically fail to take root" (Gersten and Jimenez, 438). [Also, see

McLaughlin, 1991.] Further, people learn how to usenew technologies when they are shown and taught how it will "enhance and expand" what they are already doing, rather than just insisting that what they have been doing be radically altered (Gersten and Jimenez, 438). [Also, see Richardson, 1990, and Smylie, 1988.]

Those in charge of computing resources should make students, faculty, and staff aware of the limitations and possibilities of the software and hardware available at the college. Also, they should insist that administrators and academic division heads account for and budget for training time, both on–campus and off–campus, for faculty and staff to increase software and hardware productivity.

Realistically, small colleges like Emory & Henry will probably never be able to afford to re–tool technologically often enough to stay current in all of the disciplines in which its students are interested. We can provide students with basic access to the data bases of the world—through Internet, CD roms, and other multi–media technology—so that students will be able to function at a rudimentary level. An additional step for us might be the incorporation of an externship during students' college years which might insure that students are ready to function at the levels of technological competence required in their chosen fields when they graduate.

5. Provide adequate staff to implement hardware and software.

All colleges but one in our survey felt they were understaffed to handle the resources available on campus. One contact person theorized that number of staff necessary to maintain computing technology was like a critical mass. You need a certain staff size, regardless of enrollment. Once that staff size is reached, then you can add additional staff to more completely implement the campus' computing resources.

Judging from our interviews, it seems that most small colleges have a small computing center staff in which its members often wear many hats. Time to implement new hardware and software would be added along with the jobs currently performed. Ideally, it would be advantageous to increase the computing center staff to accommodate the increase or upgrade in technology. If it is not practical from a budget standpoint to add additional people, then outside sources should be considered for installation and training to move smoothly.

Regardless of who does the implementation, the existing computing center staff needs to be adequately trained to maintain new hardware and software. After all, it will ultimately be up to them to insure that faculty, staff, and students take advantage of what's available.

An Outsider's Reaction:

Dian and Herb Thompson from Emory & Henry have already done all the hard work. Following my visit to their campus last year on behalf of the EDUCOM Consulting Group, Dian worked not only to realize the recommendations I made but, with Herb, proposed gathering the data you've just seen, to give the college a truly comparative viewpoint.

They asked me to help draft the survey instrument and to react to the results from my position at an institution similar in size and location to their surveypopulation but perhaps somewhat more fortunate in resources. Whether Washington and Lee is further down a road that these institutions are also traveling or is pursuing a strategy unique to its own situation and mission is a question only time will answer. Still, I'm fascinated by the chance to see what others are doing and thinking and by the chance briefly to air some of my thoughts in reaction.

I'll do that by answering the survey questions here, with commentary. Washington and Lee falls easily within size range of the survey, with nearly 2,000 students, of whom almost 400 are in its School of Law. Academic– and administrative–computing functions are combined in that they report to me, but with the exception of the network and operations staff (which is just being separately organized), all staff members work primarily for one or the other of the two groups, and each has a head. My visits to a number of other campuses lead me to think that the formal organizational decision is less important than the stress placed on functional integration and cooperation by the senior leadership. I've also long thought that reorganization (in either direction) might be one useful step in remedying a dysfunctional situation. Counting me, we are 13 full–time staff members, increasing to 14 this summer. This number seems slightly above average for our size, a suspicion that the Thompsons' data confirm.

Washington and Lee is at the end of a five–year plan specific to computing, and computing is part of the general strategic plan. Since with the rapid change in technology the plan can no longer be very specific about hardware and software, the principal advantage of being "in the plan" may be symbolic. It signals the senior leadership's awareness of computing's importance and their awareness that accomplishing the plan will mean devoting resources to computing.

Computing is ubiquitous at Washington and Lee, and so is the pressure for more and better resources and access. Not one department would rank computing as unimportant or their needs for it as satisfied. Every faculty member has a university–supplied computer, a network connection, E–mail, and Internet access. Every department is within the area of responsibility of a computing staff member, and where possible the offices of those staff members are outside the computing center.

We've been spending about 200 dollars per student on computer–related capital expenditures per year recently. That's a level significant enough that it's hard to pick out individual major purchases. We've extended the network to all academic faculty, and we'll be adding the administrative staff (who now communicate mainly over point–to–point connections) in the next two years. We've brought UNIX workstations and X–Terminals to campus in the last two years. Multi–media is only getting started. We've installed three new electronic classrooms with sophisticated instructional video–control systems, two with networked Dell 486 systems and one with networked Macintosh Quadras. High–speed serial connections are now present in every dorm room, as is the wiring to support full network connections when we think we're ready for them.

The biggest purchase we're considering at the moment is the replacement of our administrative systems platform, switching from a proprietary mini operating system to an open solution based on UNIX. The hardware component, interestingly, is only about a third of the cost of the project.

Like nine of the 10 schools surveyed, we train as much as we can and we'realways urging training on our clients, but (also like them) we think there's not enough.

We have about 200 micros available in student labs, most 24 hours a day, seven days a week. There's not serious pressure for more, and only during the end–of–term squeeze is there much competition. Probably two–thirds of the systems are Intel, one–third Macintosh, though that formula is drifting in the Macintosh direction. We do re–sell both Macs and Dell systems, and well over half of the student purchases are of Macintoshes. Last year for the first time, more than half of our entering students had their own computers or bought one from us by the date of fall registration. We have no formal requirement or recommendation about private computers, and we believe that university resources should ensure that no student is at a serious disadvantage, although when asked we do describe the flexibility available to students who can afford to have their own.

The campus network, fiber and 10–base–T, reaches every faculty desk and will soon also link the administrative users. Full Internet access is available to all within the university community. We belong to both EDUCOM and CAUSE, and one or more staff members attend those and several other meetings (NECC, ASCUE, ACM SIGUCCS, and vendor user's groups) each year.

The recommendations Dian and Herb drew from their survey are straightforward and sensible, and I endorse them completely. Let me close simply by paraphrasing them as negatives:

1. Don't think that computing can be divorced from the strategic goals of your institution.

2. Don't ignore what other institutions are doing.

3. Don't limit yourself to the resources of your own campus.

4. Don't assume that people will figure out how to use computing resources on their own.

5. Don't forget that hardware and software are only the raw materials of information technology, unproductive until used by human beings who will need help and support.

References Cited

Gersten, Russell, & Jimenez, Robert T. (1994). A delicate balance: Enhancing literature instruction for students of English as a second language. The Reading Teacher, 47, 438–449.

McLaughlin, M.W. (1991). Enabling professional development: What have we learned? In A. Lieberman & L. Miller (Eds.), Staff development for education in the 90s: New demands, new realities, new perspectives (pp. 61–82). New York: Teachers College Press.

Richardson, V. (1990). Significant and worthwhile change in teaching practice. Educational Researcher, 19(7), 10–18.

Smylie, M.A. (1988). The enhancement function of staff development: Organization and psychological antecedents to individual teacher change. American Educational Research Journal, 25, 1–30.

APPENDIX A

Interview Confirmation Letter

March 27, 1994

name~
college~
addr1?~
addr2?~
csz~

Dear sal~,

Thank you for agreeing to participate in our regional small college survey of southwest Virginia, eastern Tennessee, western North Carolina, and western Kentucky. As you know from my telephone call yesterday, we at Emory & Henry are in the process of trying to upgrade our computing resources. We're not exactly sure how to approach what we need to do and, therefore, are very interested in what other colleges have done or not done in our region to provide adequate computing resources for their students.

The ASCUE (Association of Small Computer Users in Education) organization has accepted our proposal for a presentation on moving small colleges into the Information Age. We know Emory & Henry is not the only small college trying to upgrade facilities and would like to include your input into problems we think small colleges are having. The questions on the following page are ones we would like to concentrate on for our presentation.

When we compile the results of our survey, we will not include specific colleges by name. The results will be used to describe generally the status of computing resources in small colleges in our region.

If you like, we will be happy to send you a copy of the survey results and/or a copy of the final paper that we submit to ASCUE.

As agreed earlier, I will call you on date~ at time~ to discuss the questions on the attached sheet. If you have any other questions or want to change the time of our telephone call, please don't hesitate to contact me.

Thanks again for your interest and cooperation.

Yours sincerely,

Dian Thompson
Computing Center Director
Voice phone:
(703) 944-4121, ext. 3839
Fax phone: (703) 944-4438

Design Issues in Computer-Based Instruction

Don Armel
School of Technology
William J. Gibbs, Stacia Kuhn
Department of Media
Eastern Illinois University
Charleston IL 61920
(217) 581-2216
cfdja@eiu.edu

Abstract

Technology educators must seek alternative instructional delivery methods. Valuable instructor time is often spent clarifying basic concepts, time that could be used for higher order learning activities such as problem solving.

For technology classes at Eastern Illinois University's School of Technology, computer-based instruction (CBI) modules are being developed. This approach was implemented because CBI is student-centered, interactive, individualized, and motivating when designed effectively.

Several instructional strategies were incorporated in the CBI design to facilitate learning and to increase learner interaction with the lesson. This paper discusses the specific methods and strategies used. Some instructional techniques such as electronic note taking and animation appeared to promote student interaction and have learning benefits.

Introduction

Technology educators must begin to look at alternative methods to deliver instruction. Valuable instructor time is often spent clarifying basic concepts, time that could otherwise be used for higher order learning activities such as problem solving. A variety of instructional approaches can be used to help alleviate this problem but few appear to be as viable as computer-based instruction (CBI).

Computer use generates interest and excitement among students (Brown, 1986). Computers integrate video, audio, text, and animation and allow for instant, simultaneous, non-sequential access to a variety of informational forms; and thus, are capable of representing knowledge concretely (e.g., video) and symbolically, (e.g., graphs) in forms with which learners can interact (Schell & Hartman, 1992). CBI is student-centered, interactive, individualized, and motivating when designed effectively.

A medium, however, is neither good nor bad and so the effectiveness of a CBI lesson is determined in part by the care with which it is developed (Hannafin & Peck, 1988). Often, the novelty of a new instructional medium has a positive influence on students (Clark, 1983), but such effects are frequently insufficient to sustain permanent educational benefit. Lasting results can best be achieved with careful, systematic planning of all aspects of the educational environment. A systems approach defines the elements that must be considered: the learner, the content, the context, the instructional strategies, and the evaluation process (Anglin, 1991). This paper discusses specific instructional methods and strategies used in the CBI design to facilitate interaction and learning.

Design Elements

A CBI approach was chosen to augment classroom instruction in graphic communication courses at Eastern Illinois University. The researchers developed a self-instructional module called Imposition which had four purposes: 1) to supplement classroom instruction; 2) to free up class time for higher order learning; 3) to make students accountable for rudimentary content instruction; 4) to make instruction more learner-centered and controlled. A design objective was to create an interface that could be used not only for Imposition but also for concepts across a wide variety of subject areas. Given this, the interface design had to be consistent. Consistent layout and design of visual information provides students stability and confidence in navigating a program (Semrau & Boyer, 1994).

Gagne's (1985) events of instruction were used to guide the development of Imposition. The instructional events are:

1. Gain attention
2. Informing learners of the objective; activating motivation
3. Stimulating recall of prior knowledge
4. Presenting the stimulus material
5. Providing learning guidance
6. Eliciting performance
7. Providing feedback
8. Assessing performance
9. Enhancing retention and transfer. (Gagne, 1985, p. 304)

For example, Figure 1 shows a screen that informs learners of the lesson's purpose and provides some advanced cues to prepare them for lesson activities. Accordingly, Hannafin and Hooper (1989) identify three components of screen design: 1) psychological, 2) instructional, and 3) technological. Information processing and perception are psychological issues. Issues pertaining to the learner or instructional content are instructional while those dealing with the potential or limits of technology are technological. Aesthetics is another component of screen design which takes into account how the combination of various elements produce favorable effects (Mukjerjee & Edmonds, 1994). The screen design of Imposition served to reinforce the concepts presented through consistency and clarity of design. As with many programs, Imposition screen designs included such elements as text, graphics and navigational tools (Haag & Snetsigner, 1994) (see Figure 1-6). The researchers consider the attributes of these elements (e.g., highlighting, placement) in terms of the aforementioned components (psychological, instructional, technological and aesthetics). For instance, button placement was done for consistency and ease of use and button icons were selected to depict functionality. Highlighting served to cue student attention and helped them make associations between the button and the action it invoked. Additionally, titles displayed across the top of the screen were intended to orient students and show the hierarchical relationships between primary and secondary concepts.

It is important to note that the accessibility of CBI authoring programs allows those untrained in graphic design to develop fairly sophisticate instructional programs. The effectiveness of these programs may be compromised, however, by the developer's lack of understanding of design elements and aesthetics. The screen designs employed in the Imposition program were the result of a cooperative effort of a design team which included a graphic designer, instructional designer, content expert and instructional programmer.

Instructional Strategies

Several instructional strategies were incorporated in the CBI design to increase learner interaction with the lesson, provide learners with on-line resources (e.g., glossary) and facilitate learning. Each of these strategies is discussed below.

Note Taking Feature

To encourage students to think about the content and make notes and to facilitate their interaction with the CBI learning process, a note taking feature was incorporated. Students had the opportunity to copy and paste text in the lesson, type notes about content, and print notes (see Figure 2). Note taking is a viable learning strategy and is supported by students (Carrie, 1983). It helps students restructure information in a scheme they understand. Allowing for the restructuring of information in ways which are meaningful to an individual can facilitate encoding and promote learning (DiVesta & Gray, 1972).

There are some factors which need consideration when building an electronic note taking function into the instructional software. From the researcher's experience, the computer affords students the flexibility to easily change and update notes. The ability to copy and paste, for example, enables notes to be duplicated and placed in lesson segments where they will be meaningful to the student. Students do, however, seem accustomed to taking notes with paper and pencil and will most likely defer to this more familiar way of working. Additionally, students with poor keyboarding skills may be less inclined to record notes than those with greater proficiency.

Displaying Text On Demand

The researchers made efforts in designing screens to ensure that the stimuli presented was purposeful and intended to influence students' information processing (Hannafin & Hooper, 1989). The Imposition module was designed for a 9 inch computer screen. Given that users want screens to be orderly, clean, and clutter-free (Galitz, 1985), much consideration had to be given to the amount of information displayed. One technique used to circumvent this limitation was to *display text on demand*. For example, the lesson presented a visual and prompted students to pass the mouse over it. As the mouse passed over important areas and labels, text descriptions displayed.

While not all the textual information was immediately available, the technique provided for clutter-free screens. Knowing that some information was hidden, students seemed to explore content in greater depth to find all the unexposed areas. As in previous work with this technique, the researchers found that students liked it (see Figure 3). *Displaying text on demand* served three functions: 1) cued student attention to important content as text fields popped-up; 2) promoted active participation of the students by peaking their curiosity and; 3) conserved screen space.

Animation

Animation as an instructional strategy has shown inconsistent results on its effect on learning (Park & Gittelman, 1992). "Theoretical and methodological problems with the use of specific attributes" of animation seem to account for the conflicting results. Outcomes tend to be positive when pedagogical principles are followed as animation is designed into instruction. Animation is most effective with abstract and dynamic information. It is also effective for gaining attention (Wilson, 1993) and maintaining motivation. It is best used to simulate behavior and represent movement, and it facilitates explaining of complex knowledge (Park & Gittelman, 1992).

One strength of computer-based instruction is the ability to incorporate animation in lessons. The Imposition module presented animated sequences representing specific processes related to imposition - a graphic arts process in which pages are laid out for a press form so they will be in correct sequence when folded and bound. In one exercise, students folded a piece of paper to simulate an imposition form and to become oriented to fold direction and page numbering. They could review the animation to verify the accuracy of their models (see Figure 4).

Glossary

To understand basic imposition concepts, students must be familiar with imposition terminology and definitions. To accommodate this need, an on-line glossary of terms was available throughout the module (see Figure 5). The glossary was valuable for several reasons. First, students could use it to instantly find what they needed. A design objective was to eliminate all obstacles which interfered with students' ability to access required information. Second, the content relied heavily on terminology and definitions. Therefore, students first had to have a working knowledge of imposition terms to understand fundamental concepts and processes.

Comments

When using the module, students could type comments; a feature which gave them the opportunity to express their likes and dislikes. Student comments are valuable during formative evaluation and can be used to refine and improve the instructional software. Thus, collecting student critiques was used during the development evaluation trials and turned out to be a good way to anonymously collect needed information about the design of the module. Additional benefits of allowing students to comment on the instruction are: 1) commenting may help students develop a sense of ownership for the instructional material, 2) commenting may involve students with instructional development and afford them a sense of learner control. Because of its value as a feedback vehicle, the comments feature will likely be included as a permanent program option.

Summary

One reason for choosing computer-based instruction was to introduce a new paradigm for teaching technology at Eastern Illinois University, as well as to use class time more effectively and to promote problem solving. An objective for the Imposition module was to develop a interface framework that could be used in many situations and subject areas. The uniformity of the framework provides a level of consistency that students can depend on as they move from module to module and across content areas. In the researchers' view, the Imposition module included some instructional techniques such as electronic note taking and animation which appeared to promote student interaction and facilitate learning.

Bibliography

Anglin, J. (Ed.). (1991). *Instructional technology: Past, present and future.* Englewood, CO: Libraries Unlimited, Inc.

Brown, J. (1986). Some motivational issues in computer-based instruction. *Educational Technology*, 26(4) 27-29.

Carrie, C.A. (1983). Notetaking research: Implication for the classroom. *Journal of Instructional Development*, 6(3), 19-26.

Clark, R. E. (1983). Reconsidering Research on Learning from Media. *Review of Educational Research*. 53(4), 445-459.

DiVesta, F. J. & Gray, L. (1972). Listening and notetaking. *Journal of Educational Psychology*, 63, 8-14.

Gagne, R. (1985). *The conditions of learning*. New York: Holt, Rinehart and Winston.

Galiz, W.O. (1985). *Handbook of screen format design*. MAL QED Information Sciences Inc.

Haag, B. & Snetsigner, W. (1994). Aesthetics and screen design: An integration of principles. In Beauchamp, D., Barden, R. & Baca, J. (Ed.). *Visual Literacy in the Digital Age*. Blacksburg, VA: The International Visual Literacy Association, Inc.

Hannafin, M. J. & Hooper, S. (1989). An integrated framework for CBI screen design and layout. *Computers in Human Behavior*, 5, 155-165.

Hannafin, M. J. & Peck, K. L. (1988). *The design, development, and evaluation of instructional software*. New York: Macmillan Publishing.

Mukherjee, P. & Edmonds, G. (1994). Screen design: A review of research. In Beauchamp, D., Barden, R. & Baca, J. (Ed.). *Visual Literacy in the Digital Age..* Blacksburg, VA: The International Visual Literacy Association, Inc.

Park, O. & Gittelman, S. (1992). Use of animation and feedback in computer-based instruction. *Educational Technology, Research and Development*; 40(4), 27-38.

Schell, J.W., Hartman, D.K. (1992). Flexible Cognition Theory: A framework for advanced vocational learning. *Journal of Industrial Teacher Education*, 29(3), 39-59.

Boyer, B. & Semrau, P. (1993). Selecting and evaluating interactive videodisc programs in art. *The Computing Teacher*, 21(3), 28-30.

Wilson, L. (1993). Get moving. Animation brings classroom visuals into the '90s. *Vocational Education Journal*; 68(2), 30-31.

The Impact of Telecommunications on Education: Some Serious Considerations

David J. Ayersman
Daniel K. Anderson
West Virginia University
Morgantown WV 26506–6122
djayers@wvnvm

Abstract

There are many question raised by the use of computer-communications; particularly within education and government. What exactly is the information highway as touted by the Clinton Administration? Does this technology broaden or narrow access to information? Are there significant barriers (i.e., socio-economic, geographic) that might limit an individual's access to information? How might it impact education? What will be some of the ethical dilemmas presented by this new technology? This paper will begin to explore some of these issues and will provide an information source and a perspective to be considered by educators and administrators.

Introduction

In September of 1993, the White House announced a plan to establish the Internet as the starting point for the National Information Infrastructure. The Clinton administration has touted the National Information Infrastructure as an "information highway" that will redefine communications. As educators and citizens we are obligated to become more informed of these electronic possibilities so that we may avoid the potential downfalls associated with them.

Government policies that were developed to regulate and ensure the flow of information have largely been established with print-forms of information in mind. Current technological capabilities require us to re-think these policies as new forms of information are established. If there will be regulatory laws governing such a plan, it would seem beneficial to anticipate them and educate ourselves on these issues so that we might be better prepared for them.

The audio-visual media_radio, television, and films_broadened access to information by making it cheaper and less demanding of skills, but reduced the opportunities for public input since the information was primarily limited to presentation of information to a passive public. Telecommunications offers the ability of information-exchange allowing the public to become active participants in the flow of information. With the advent of teleconferences and telecommunications, the role of theadministrator has changed. No longer is he or she to be found in the midst of a shield of clerical workers and staff. Administrators are now directly accessible via an e-mail address or a fax number.

There are many question raised by the use of computer-communications. Does the technology broaden or narrow access to information? Does the economic organization of the newer media give private owners, as gatekeepers, a power to deny or limit access? Do public authorities attempt to deny or limit access? How much will access vary from place to place and school to school? Are there significant barriers that limit the access of individuals to information? Who should pay the costs and how will this affect access? What will be some of the ethical dilemmas presented by this new technology?

It should come as no surprise that new technology offers both advantages and disadvantages. Unfortunately, when we fail to consider some of the negatives involved with change we often neglect to prepare ourselves for the eventual impact these negative outcomes might have. A goal of this paper is to identify and describe issues relevant to telecommunications and education so that they can be considered and discussed as a preliminary step leading to their eventual resolution. This paper will begin to explore some of these issues and will provide an information source and a perspective to be considered by educators and administrators.

Computers and Telecommunications

The advent of computers following World War II has significantly impacted our society. From monolithic pieces of equipment to current hand-held computers, advancements have steadily been made. Information storage capabilities have changed tremendously with the introduction of optical storage devices able to store entire volumes of information on a single compact disc. The speed at which computers process information has continued to increase at such a rate that it is difficult to remain "current" with the latest hardware. Prices have continued to decrease as more efficient and less expensive methods of computing have evolved. One of the most recent developments in computing is telecommunications or computer-communications. Telecommunications is the process by which computer users communicate with each other and gain access to services across telephone lines or other types of public data links. Telecommunications opens a new world of information access and exchange. Currently there are videophones that allow persons to view the person to whom they are speaking (just like the Jetsons). Teleconferencing can incorporate many aspects of live video, sound, and interactive conversations involving multiple persons separated by geographic distances. The infrastructure for these types of communication-he international network of telephone and data links (internet)-is already in place and ready to be used.

The Internet (previously known as the ARPANET) was established in the mid-1960's, during the cold war, by the Department of Defense. The Internet was intended to provide communication alternatives to telephone, radio, and TV for military purposes in the event of hostilities with the Soviet Union. Internet has grown from the government's ARPANET (Advanced Research Projects Agency Network-Department of Defense) and the NSFNET (National Sciences Foundation Network) to consist of over 20 million users while adding a million more every month (Elmer-Dewitt, 1993). The Clinton administration has formed the Information Infrastructure Task Force (IITF) to deal with issues relevant to the Internet.

Paul Baran, a researcher for the Rand corporation in 1964, developed the design for a communications system that had no hub, no central switching station, no governing authority, and that assumed every link was unreliable. Rather than being an orderly system of precise paths of communication, the Internet was intended to achieve operational success via multiple pathways within an overall duplicitous framework, that by its very nature, would be continuously operational since no single pathway was integral to the flow of communication. This has become the basis for the Internet and computer networking in general. Elmer-Dewitt (1993, p. 62) provides a general explanation of how the Internet operates:

In Baran's scheme, each message was cut into tiny strips and stuffed into electronic envelopes, called packets, each marked with the address of the sender and the intended receiver. The packets were then released like so much confetti into the web of interconnected computers, where they were tossed back and forth over high-speed wires in the general direction of their destination and reassembled when they finally got there. If any packets were missing or mangled (and it was assumed that some would be), it was no big deal; they were simply re-sent.

To successfully communicate on the Internet requires (a) access to a computer capable of communicating on the Internet and (b) some mastery of basic computer skills_language and commands. Corporate agencies, government, and individuals are scrambling to acquire the resources and skills to develop this capability to communicate.

Who pays for the Internet? Although the government initially provided the majority of funding, they are beginning to cut back (two years ago the National Science Foundation lifted restrictions against commercial use of the Internet) and funding is now being provided by other sources. The major costs of running the network are cooperatively shared by universities, corporate agencies, and foreign governments. Currently there are few laws or regulations imposed for those on the Internet, but there are general standards of acceptable conduct that for the most part, are adhered to by the users, making it a truly democratic mini-society in that they are self-ruled.

The number of schoolteachers knowledgeable about internet is rapidly growing (Jacobsen, 1994) and now includes tens of thousandsof teachers in elementary and secondary schools. Many colleges and universities are making their links free to schools. The High-Performance Computing Act of 1991 included a component which established the National Research and Education Network (NREN). Purportedly, the NREN was to fund the enlargement and updating of the Internet, the linking of educational institutions to the net, the creation of more robust standards and less cryptic commands for net navigation, and the research necessary to implement these goals. A number of computing organizations sent requests for comment to the task forces which were establishing plans for the NREN. For example, Mitch Kapor, creator of Lotus 1-2-3 and President of the Electronic Frontier Foundation, called for the NREN to "rely on open technical standards whose specifications are not controlled by any private parties and which are freely available for all to use" to ensure access for new information providers and to prevent the establishment of a monopolistic computing environment. EDUCOM President Ken King sent letters to government agencies requesting legislation that would facilitate the "creation of a federal, state, and local networking partnerships, with contributions from all levels" which he felt would be essential to the success of the network.

The primary backbone of telecommunications systems in the United States, Internet, is analogous to a system of highways. Internet is the interstate while smaller networks linked to Internet are comparable to secondary roads or blue highways (the color depicting smaller roads on a map). A second analogy is that a communications network is similar to a sieve in that no matter which hole you plug there is still water running out_or, in this analogy, information reaching its destination.

Connecting these numerous networks are "gateways" and providing services is accomplished through them. For example, if someone were to want access to Internet from

West Virginia University they would likely go through BITNET or WVNET (West Virginia Network for Educational Telecomputing) to get there. They would require an account through an agency or organization (like WVU) that would provide them an account number and the use of their equipment as a host. Private citizens can access Internet with a computer and a modem if they have an account with an affiliate or some type of commercial service for permitting access.

Currently, there are few printed guides to follow when using Internet. This is likely to cause new users to experience difficulties in navigation as they become disoriented in a sea of information resources that remains hidden and beyond the horizon to them. There are many recognizable advantages of computer-communications networks however. Using networks to disseminate information to the public provides information that is timely. The numerous sources of information allow the user to explore a variety of topics. Lifelong education can become a reality for more of society as we become solely responsible for finding what we are interested in while not being inhibited from using the technology to do it. Gathering information about a topicof interest can further our interest in that topic or create new topics of interest along the way. There is the possibility of enhanced social cooperation as more people begin to communicate and they do it more often. One aspect of this increased communication might be more political interaction that can be afforded by networking. There is no limit to the number of topics that people might discuss. As people realize their own abilities to voice opinions and contribute to information exchanges, the town-meeting model, as proposed by Ross Perot, in which groups of people can deliberate directly and engage in decision-making, looms ever-closer.

Democracy

In a video titled "Counting on Computers", Walter Cronkite argues that computer use by its very nature leads to the development of a scientific autocracy_a threat to democracy that can only be countered by fostering greater scientific literacy at every level in our schools. There are essentially two types of democracy. In a direct democracy, every member of a group has an equal right to take part in all of the group's decisions_that is, the whole group decides what to do; its members do not elect representatives. In a representative democracy the citizens choose individuals to speak and act on their behalf. Despite popular belief, our government is not intended to be a government run by the people. It is a government by an administration under authority of the legislature, with the consent of the governed, with this consent being given only at intervals and never being unanimous (Coyle, 1940). Similar to something once said by Rousseau, in a representative democracy the people are only free while they are voting for representatives to government. The trend has been to elect representatives to office (generally with an apathetic voter turnout allowing the electoral college to actually decide who the representatives will be for presidential elections), then to spend the duration of the elected official's term in office criticizing their actions while doing little else about it. In a direct democracy, the people represent themselves and issues are subject to a great amount of deliberation with so many people being directly involved. There are similarities to plenary review inherent in this type of democracy as one can foresee a continual need to reach decisions by forcing a vote rather than allowing issues to wither and die through prolonged deliberation.

Few individuals have the time needed for such a democracy. We cannot remain up-to-date on all the issues all the time and still lead productive lives. One option has been to rely on representative democracy. A second option would be to rely on those persons able to remain active in government to carry the weight of deliberation so that issues can be boiled down to votable options for the citizenry. If citizens were able to tune in to the deliberation at their whim, ideally we would have a better informed citizenry making decisions that they would then become owners of. Ideally that is. But as someone once said, "the most democraticprocedure is the very one most likely to change that procedure."

Information and Democracy

Information is the currency of democracy.
_Ralph Nader

By the year 2000 high school seniors will come into contact with as much information in the course of a year as their grandparents did in the course of a lifetime (Cetron, 1988). There is little doubt that we are in the midst of an information explosion. No longer can we hope to know everything about a topic, we are resigned to knowing how to access everything we can about the topic instead. As educators and administrators who might read this breathe a sigh of relief, it might be beneficial to remember that this does not remove the responsibility of being aware of policy and constitutional themes which affect how the public is served.

There are essentially two aspects of information within a democracy and they are the people's right to know and their rights to privacy. Oddly, these two issues can be conflicting and are often misunderstood. Educators and administrators should have a working knowledge of both issues for their own protection as well as to enhance their ability to serve the public. The government is often in the business of providing information to its citizens. The nature of the information to be released must not conflict with individual rights of privacy nor must it conflict with security of government. The Freedom of Information Act (1966) and the Privacy Act (1974) address these issues and will be discussed shortly.

It is generally accepted that information is the source of knowledge and knowledge is the source of power. Within a democracy it is imperative that the citizenry have free access to information so that a government run "by the people" can be run "by an informed people." Information can be a source of social definition within a democracy. The groups that we affiliate with, the causes we support, and the stances we take in discussions are all a result of using information to define ourselves socially. Distributing information within a democracy has created many problems. One recent suggestion (made popular by Ross Perot) to enhance the distribution of information by government is the use of town meetings, or town meeting halls, as electronic information distribution sources. It is hoped that democratic participation in decision-making might be enhanced by the creation of facilities such as these, and other means of computer-communications, that will lead to a realization of the democratic ideal itself (Gould, 1989). The hope is that "electronic democracy" will increase participation in government. Unfortunately, the people least involved in government now are those least likely to afford the electronic democracy.

Computer networking within a democracy has many uses, from finding out what people think to counting votes. It can provide access torepresentatives and information otherwise not readily available. To facilitate democracy in this type of environment calls for broadening and cheapening communications for public use. Videophones and interactive TV are no longer on the horizon, they have arrived and as the prices drop they will become more accessible to the public.

The moter-voter bill that recently died in Congress, would have allowed citizens to register to vote as they applied for driver's licenses. To carry the idea a few steps further we can imagine drive-thru polls on election days where the voter provides the necessary identification, is provided a ballot, and then proceeds to vote without getting out of the vehicle. To go still further, we can imagine that if we were able to provide the appropriate identification (an assigned number or address of some sort on file with the FCC [Federal Communications Committee]), we could access the polls electronically via computer-communications, pull up a ballot (perhaps with our address already on it), and proceed to vote without ever having to get out of the house. In less serious matters than national elections computer-communications have already been used for voting purposes. One might be concerned that there would be no assurance of one-person one-vote, but there are ways of avoiding computerized "ballot stuffing", such as audits that use random identification numbers (Gould, 1989).

Public's Right to Know

> Knowledge will forever govern ignorance: And a people who mean to be their own governours, must arm themselves with the power which knowledge gives. A popular government without popular information, or the means of acquiring it, is but a prologue to a farce or a tragedy; or perhaps both.
> _ James Madison

There are some who argue that the public's right to know is an indirectly secured right originating in the First Amendment's freedoms of speech and of the press (O'Brien, 1982). The public's right to know is not something explicitly described in the Constitution however. Because of this lack of historical basis for the proposition that the First Amendment was intended to secure this individual right to demand access to government information, the claim of First Amendment origins is a pretense with no authentic basis (O'Brien).

Some, more liberal, Justices of the Supreme Court in the 1970's began to interpret the First Amendment as embodying an affirmative "right to know" and that a special privilege is granted to members of the press as the agents acting on behalf of the citizenry (O'Brien, 1982). The press is recognized as serving as a guardian of informed public opinion through their attempt at complete coverage of news events. Because of this, for some, freedom of information is defined as freedom of the press.

The Freedom of Information Act (FOIA) was passed by Congress in 1966 to assist the public in obtaining information from government agencies. Amendments have since been added that have continued to facilitate this information access to the public. With the passage of the FOIA, the burden of proof passed from the individual (who previously had to justify a "need to know") to the government (who now recognizes that the individual has a "right to know"). The FOIA applies only to documents held by administrative agencies of the executive branch of government. There is no First Amendment right of access for the public or the press to judicial or other governmental proceedings (Justice William Rehnquist, Gannett Co. v. DePasquale, 1979).

The Justice Department oversees how the federal agencies respond to the requirements of the FOIA. The FBI, CIA, and many corporate businesses reacted negatively to the openness of the act and requested more exemptions to reduce the burden of providing information. Other reasons to resist the FOIA involve the cost of providing the public information. The estimated cost to the taxpayers in 1980 was $50 million (WANT, 1984). Even more reasons are available if one considers that about 15% of the FOIA requests come from incarcerated individuals. One perspective might be that the incarcerated individuals are attempting to gain evidence to support their case while another is that they might be attempting to identify the informant that is responsible for putting them there_either way, these requests are at the taxpayers' expense. Despite the openness of the original FOIA, there has since come the realization that even in a democracy some secrets are needed and this is evidenced by nine major categories of exclusions for the act.

The FBI in particular, has not welcomed the obligations imposed upon them by the FOIA. Considering that Hoover was still Director for six years following the passage of the act, it is little wonder. It has been more recently recognized, however, that the FBI has improved its responses to FOIA requests (Lowe, 1978). The FBI's hesitation to uphold the FOIA is similar to a previous attempt at restricting information access to the public_the Library Awareness Program.

This program was a counterintelligence activity that began in the 1960's under Hoover and continued into the 1980's under the most recent Director_William Sessions (Foerstel, 1991). The program involved FBI agents attempting to gather information on users of public libraries through librarians and library records in an effort to establish and monitor "suspicious" behavior on behalf of domestic intelligence.

In 1974, the Privacy Act was passed by Congress. The Privacy Act is intended to provide individuals the ability to acquire information about themselves and to prevent disclosure of this information to others. This act gives the individual significant control over how information about oneself is used. It requires the individual's consent prior to the release of information abouthim or her and prescribes that no secret record systems are to be kept on individuals. J. Edgar Hoover would have had difficulty in adhering to such a system-moreover, he died in office on May 2, 1972 (Summers, 1993).

There is a dichotomy of the public's right to know and the obligation of the government to maintain confidentiality. It is addressed by the FOIA/Privacy Acts and the failure to disclose must be specifically justified within one of the nine exemptions to the FOIA.

Ethical Dilemmas

There are many areas that pose ethical dilemmas when using computer-communications. Educators and administrators are particularly affected by dilemmas that question the Bill of Rights and the two articles of legislation previously mentioned_Privacy Act and the FOIA. There are specific rights of protection provided through the First, Fourth, Fifth, and Sixth Amendments that raise important issues for consideration. Administrators would benefit by thinking about these issues before they present themselves as unanticipated problems. Ensuring information and services access and availability to the public raises a second series of issues relating to the Privacy Act and the FOIA.

Lacey (1986) recognizes that new technologies go through a three-stage process: (1) initially the technology is available to only a few, and its possession grants them power, (2) it spreads and becomes a democratizing force, and (3) it is so common that to do without it puts one at a distinct disadvantage. If these steps are accurate, we need to consider how computer-communications might affect education by examining specific contexts of ethical issues. By looking at specific amendments and legislation that will be affected we can achieve this.

First Amendment

> _Congress shall make no law . . . abridging the freedom of speech, or of the press; or of the right of the people to peaceably assemble. . .

With computer-communications, there is a broader range of access for expressing one's thoughts. If I knew President Clinton's e-mail address, I could advise him on issues of international significance. He couldn't stop my electronic mail from reaching him regardless of what I might offer as advice nor could I stop the potential onslaught of commercial "junk-mail" that might be created as a result of the NSF removing commercial restrictions from the Internet. Will there be some balance between freedom of speech and freedom to ignore everyone's else's speeches?

It takes only two persons to communicate via Internet. Recently Iheard of a course offered on the Internet that was attended by over 1700 students from around the world. The potential to gather en masse is an obvious advantage of computer-communication. Will this right be in jeopardy as gatekeepers (possibly government affiliated) decide who gets access to the network? What about national/international communications? There is no passport required to journey throughout the world via Internet. Is this a regulation in waiting? If inmates are communicating via Internet, are they really being confined according to the intentions of the courts?

Despite the fact that the press is charged with the duty of informing the people, votes have been cast on the basis of media-controlled information to the public that has not represented the entire issue. Less control over the information might be one advantage of computer-networking. Sorting out the truth from the opinions might be the downside of this however.

The anonymity available through computer-communications allows us to speak freely and without reservation. If we are aware of unethical or inappropriate acts being committed and wish to do something about it, by confronting the guilty person (if they are higher than us in the organization) we face the possibility of dire consequences. It is not uncommon to hear of litigation attempting to settle disputes of whistleblowers who have been fired from their jobs. Lodging a complaint through computer-communications allows us to reveal the facts without fear of losing our jobs or of even being identified as the source of the complaint. Admittedly, speaking out openly might be much more convincing, but often just to reveal the information to the appropriate persons is all that is needed. Benefits of this capability are the immediacy of the complaint reaching its destination, the far-reaching capacity of being able to send it almost anywhere, and the anonymity that it offers. Disadvantages might be that filing a complaint in this manner wouldn't carry the same weight as a face-to-face revelation of the facts where you are putting much more on the line. Perhaps this will encourage government employees, and others, to act without hesitation in ethical crises that warrant a whistleblower. This could have the result of forcing government to run more openly so that more is known about its operations. Will information on Internet flow under the protections of the First Amendment? This remains to be seen.

Fourth Amendment

> _The right of the people to be secrue in their persons, houses, papers, and effects ...

There is practically no security against computerized invasion of one's electronic communications. Using someone else's password or address can enable someone to move about with an electronic "disguise" that would be the envy of James Bond. I can access information belonging to others just as easily as others can access mine. Government agencies are no exception. The NSA, FBI, FCC,and possibly the CIA are all capable of monitoring the Internet. There are apparently no restraints preventing this round-the-clock observation. Will Fourth Amendment violations become a problem? Will the Fourth Amendment apply to government's virtually undetectable intrusions via telecommunications? Will the government act as monitors of bulletin boards (those who are charged with the discretion of becoming a gatekeeper of information on computer networks)?

In an area virtually free of regulation (computer-communications), there is the appearance of a need for a central authority to step up so that order can be established. Assistance can be provided to users, regulations can be established to prevent unacceptable behaviors, and a general overwatch of the system could then ensue. Currently the nature of the Internet is seen as a free environment in which no one owns the net and the users operate freely and independently. I doubt government intervention will be very welcome.

Fifth Amendment

> _No person shall . . . be deprived of life, liberty, or property without due process of law; nor shall private property be taken for public use, without just compensation.

What constitutes property within the realm of computer-communications? Software is made-up of binary code and Boolean logic that is further complicated by units of information so small (bytes) that there is no clear solution to defining property or establishing ownership. Copyrights and patents are time-consuming endeavors normally devoted to printed material and some commercial forms of software. Computer-communication involves passing rough drafts and raw forms of materials across great distances in a timely manner inconsistent with the copyright and patent processes. How will we ascertain when property has been taken for public use without just compensation?

Sixth Amendment

> _In all criminal prosecutions, the accused shall enjoy the right to a speedy and public trial, by an impartial jury of the State and district wherein the crime shall have been committed . . .

Internet involves sending information in multi-linear routes consisting of circuitous paths that cross state, district, and international boundaries in mere seconds without the knowledge of the sender or the receiver. Will computer-communications crimes be automatically classified as federal crimes because of this? If so, the seriousness of the crime would seem to be escalated simply because of the jurisdiction. The origin of the criminal misbehavior or perhaps the route it takes are two options to consider. How will the courts determine venue?

Privacy Act

Computer-communications may require redefining privacy now that there are no traditional barriers in place. To prevent disclosure of information about individuals would seem to require some considerations of security regulations for the realm of computer-communications as more governmental agencies consolidate information databases.

The Privacy Act specifically states that no secret record systems will be kept on individuals. The FBI is charged with domestic intelligence and has a prior history of such record systems (primarily under Hoover). Will the substance and intent of the Library Awareness Program_to establish secret records based on publicly accessible information_be replicated by the National Security Agency (NSA), part of the Department of Defense, who is charged with communications intelligence and security activities? Or might the FBI, in the domestic intelligence realm of government security, attempt to monitor the Internet for spies and criminals? Admittedly, the computerized sophistication of the "new" FBI provides a much greater potential for abuse than anything possible during Hoover's reign (Poveda, 1990).

The current director of the FBI, Louis Freeh, spoke recently (December 8, 1993) on C-Span to members of the Free Press International. In his presentation he called for legislation that would allow agencies, like the FBI, access to the new forms of communication (computer-communication). Without this access he stated that information or evidence normally gathered through wiretaps would be freely transmitted through these virtually unmonitored means.

Without external checks and balances on the FBI and the Department of Justice, there is a real possibility that they will cast a much wider net than necessary and that it will ensnare innocent members of the community in their efforts to detect crimes of hard-core criminals. The ease of access for information possessed by the FBI is one of the dangers of the "new" FBI (Poveda, 1990).

Freedom of Information Act

The Freedom of Information Act has resulted in the release of enormous amounts of information from government control. It has reduced the amount of government censorship and has thus resulted in a more open government. It has not gone unopposed and in fact there have been times that government has made decisions that seem to be contrary to the intentions of the FOIA.

The Paperwork Reduction Act of 1980 (Public Law 96-511, 94 State 2812, Chapter 35 of Title 44 USC) established a broad mandate for agencies to perform their information services in an efficient, effective, and economical manner. One interpretation of the impactof this law was that information was being provided less and less by the government to its citizens. The ambiguity of the wording was partially to blame as the director of the Office of Management and Budget became the authority to interpret the law. Although the original intention was to reduce duplication and waste within government (an admirable goal), the effect was to limit information availability. Since information availability is the common theme of the FOIA, there are several specific areas that will be affected by computer-communications.

Information Organization. As on-line information repositories categorize, code, and sort in an attempt to meaningfully organize the information on hand, users will most likely continue to unintentionally exclude information from their search due to the organization of the information. Which words are to be used as keywords or subject headers to identify an article or book? On-line systems may exacerbate these problems as much more goes on behind the scenes and information is more narrowly defined. The effect will be that organizers of information databases may contribute to the government's unintentional limits on information availability to the public.

The form of information_electronic as opposed to print-form_may prove to limit access. There is already some information that is available only in electronic form (Lacey, 1986). When television was first introduced, there was a concern that many newspapers would go out of business. In fact, the sale of newspapers did initially decline but has since recovered. Will the introduction of computer-communication follow a similar pattern by initially reducing print-forms of information or will the trend continue without recovery?

Geographical Inequities. In the past, the public library system has responded to rural citizen's needs by taking the library to them (i.e., bookmobiles and branch libraries). President Clinton's government appears to be following the same approach by suggesting that downlinks and terminals will be available in public buildings for similar reasons_geographical remoteness. Most rural areas lack the resources found in urban libraries. Making information available on-line creates the apparent possibility of solving this crisis. Will rural areas have the same types of terminals (hardware and software) as the urban areas? Will librarians or resource personnel have similar abilities of providing assistance at the two locations? Will citizens someday check out portable computers for this purpose just as they initially began to release books for public circulation? These questions raise some interesting issues to be considered.

Geographical distances are not limited to just the United States. The world is now much more international and governmental information is needed in countries without sufficient computer-communications resources. While there are countries that still have paper shortages (Tanzania, East Africa) it seems ludicrous to suggest that computers can alleviate informationdeprivation in the Third World. Rural areas throughout the world are information-poor. Sources of information depending on radio and TV frequencies are not spread evenly throughout the country let alone the world. Will the blue highways of Internet suffer a similar rural versus urban inequity that results in limited access for rural areas of the world?

Socioeconomic Disparities. Computer-communication may prove to add the lack of information to the tribulations of the poor (Blake, 1986). The cost of accessing information via telecommunications and computers is higher than many citizens can afford. Many lack the ability to make use of these opportunities. Institutional access in the form of computers in public libraries and universities may pose a solution but channeling information through a few isolated terminals forces the public to individually gain access in the form of time on the terminal. This is opposed to multiple users simultaneously browsing freely through a library stocked with numerous books.

These new information formats have served as an excuse for contemplating user fees which raises the issue of government as provider and disseminator of information and subsidizer of the cost (economic considerations) of this information. Will government be able to pay for these kinds of information resources?

The cost of computers_much higher than traditional print-form materials_*may result in a limit on information availability.* The cost of maintaining the Public Broadcasting Channel is frequently debated in Congress. What will be its response to providing libraries and other government buildings access to Internet and other forms of telecommunications? Simply put, computer-communications systems are one reason the distance between the poor and the rich is continuing to grow (Lavery, 1986).

Physical Ability. The lack of ability or skill to use the media is recognizably another barrier to information availability. There are many groups of society that might have difficulty using computer-communications just as there are those that will become even more able. If mobility is a problem, computer-communications will enable more people to overcome it since they can gain access to information (and to government) directly from their homes. Visually-impaired citizens could communicate via voice-controlled programs that alleviate the need to read text. Other groups of citizens might be put at a disadvantage. The learning disabled and illiterate would seemingly have difficulty. The U.S. Department of Education, in 1987, estimated that there were more than 27 million adult functional illiterates in the United States alone (Lacey, 1987). Although select groups might benefit, attention needs to be given to other groups that will be put at a disadvantage.

Legal Issues

There are many legal issues that arise as a result ofcomputer-communications. When new technologies are evolving far more rapidly than the legal system can handle, there are bound to be problems (Gemignani, 1985). It is not difficult to mask one's identity by using someone else's account number or password. This creates issues of libel and defamation that the courts will have to address. Proving these deliberate acts will be difficult. There are copyright and licensing issues created by the ease of transmitting and making multiple copies of programs.

Specific professions will face their own ethical dilemmas as a result of telecommunication systems. Lawyers face issues of lawyer-client communication security, problems of non-lawyer access to the system, difficulties in abstaining from providing advice to clients outside their geographic area of expertise, and the anonymity of clients may result in conflicts of representation. One could extrapolate these problems to other professions medical advice given by unqualified individuals and educators releasing grades to individuals other than students, for example.

Just as computer-communications might result in a more open government as more people act to reveal the truth, it can also result in more people using the same methods to slander and misrepresent the truth. If the libelous information becomes the basis for managerial decisions, I foresee a move toward a more litigious society as people seek recourse.

Summary/ Conclusions

To make the information available to all requires (a) an infrastructure linking the many electronic sources and (b) enabling individuals to use the new technology. Internet largely fulfills only one of the two requirements. The other remains to be addressed.

Using the Internet to provide information to the public only solves the first half of the information problem. The other half, not yet dealt with, will be providing users the ability and access to use the Internet. Without adequate education or knowledge of how to use the system there will be an inability to gather and organize the available information all around us. There are specific groups of people who will suffer from a reliance on computer-communications. If the needs of these groups go unnoticed there will be a collection of illiterate, impoverished, and information-deprived sufferers who remain out of contact with the mainstream of society.

The Internet has flourished so far without much regulation from government. The need for some type of central authority to make an attempt at providing some control over the Internet is a possibility that government will begin to regulate more of the computer-communications world. "While it may be difficult for communities as diverse as those on the Internet to set their own agenda, it seems increasingly likely that if they don't, someoneelse will do it for them" (Elmer-Dewitt, 1993, p. 64).

There is little doubt that telecommunications and computers are increasing the diversity and quantity of the information flowing around the world. There is a need to more specifically examine the impact of this information flow. Who is participating? Who is not? Is the distance between the information "haves" and the information "have nots" spreading because of it or is it narrowing? Some of these issues, such as privacy, intellectual property rights, and information security are already being addressed by the committees and working groups of the IITF. Other issues, such as user acceptance and organizational learning, still need to be addressed by the IITF.

James Madison had to be convinced a Bill of Rights was necessary. He believed that the real source of danger to people's freedom would come from the majority of people ruling over those who were in the minority, not from government. If the well-to-do who have computers and access to information through systems like Internet are able to participate in government, the minority will continue to be neglected and left out of the democratic process.

References

Blake, F.M. (1986). Information and poverty. In J.P. Lang (Ed.), Unequal access to information resources: Problems and needs of the world's information poor. Ann Arbor, MI: Pierian Press.

Cetron, M.J. (1988). Class of 2000: The good news and the bad news. The Futurist, 23(6), 9-15.

Coyle, D.C. (1940). Information. In M. Marx (Ed.), Public management in the new democracy. New York, NY: Harper & Brothers.

Elmer-Dewitt, P. (1993). First nation in cyberspace. Time, 142(24), 62-64.

Foerstel, H.N. (1991). Surveillance in the stacks: The FBI's Library Awareness Program. New York, NY: Greenwood.

Gemignami, M. (1985). What are the legal rights and responsibilities of the providers of on-line services? Abacus, 3(1), 54.

Gould, C.C. (1989). The information web: Ethical and social implications of computer networking. Boulder, CO: Westview Press.

Jacobsen, R.L. (1994). Connecting the schools: Colleges join efforts to bring Internet access to elementary and secondary classrooms. The Chronicle of Higher Education, XL(22), A17-A18.

Lacey, D.M. (1986). Freedom and equality of access to information: A report to the American Library Association. Chicago, IL: American Library Association.

Lavery, M. (1986). The information age in a feudal society: BASIC literacy versus basic literacy. In J.P. Lang (Ed.), Unequal access to information resources: Problems and needs of the world's information poor. Ann Arbor, MI: Pierian Press.

Lowe, V. (1978). FBI compliance with the Freedom of Information Act. Hearing before a subcommittee of the committee on government operations, House of Representatives, 95th Congress. Washington, DC: U.S. Government Printing Office.

O'Brien, D.M. (1982). The public's right to know: The Supreme court and the first amendment. New York, NY: Praeger.

Poveda, T.G. (1990). Lawlessness and reform: The FBI in transition. Pacific Grove, CA: Brooks/Cole Publishing Company.

Summers, A. (1993). Official and confidential: The secret life of J. Edgar Hoover. New York, NY: G.P. Putnam's Sons.

WANT. (1984). Freedom of information guide. Washington, DC: WANT Publishing.

CWIS Development from Beginning to End

Eric Gallagher
Christine Mathews
Hood College
401 Rosemont Avenue
Frederick MD 21701
301–696–3907
301–696–3568
egallagher@nimue.hood.edu
mathews@nimue.hood.edu

Introduction

In the fall of 1992, the Career Center received a grant from Mrs. Catherine Filene Shouse. The purpose of the grant was to create an electronic employment center that would allow Hood College to perform a service for the entire world. But the Shouse grant did more than that. It provided Hood with the critical material necessary to bridge the technology gap existing between the campus network and a true Campus–Wide Information System (CWIS).

Network History

Of course, the CWIS project began long before the Shouse donation made it possible. Hood is a school of only two thousand students and the physical campus is correspondingly small. Our size allowed the Computing Affairs department to respond quickly when the need for a physical computer network arose.

To move from a series of isolated labs to a campus network, we laid fiber–optic cables underground from building to building. We had to keep finances in mind, however. The Computing Affairs department designed the network with the help of the Digital Equipment Corporation. The goal was to build a network that was capable of expansion but made use of our present DEC equipment. Although the backbone of the campus network is fiber cable, buildings themselves were wired with unshielded twisted pair. The fiber runs only into a few high–performance locations.

We designed the office hubs in each building to be capable of running either LAT or 10-Base-T. LAT connections gave Hood an easy, inexpensive way to connect PCs to a network. (Ethernet cards are not required for such a network.) However, the connection is slower than ethernet. Eventually, we will want to grow into something faster – and, with the hub arrangement, it will be easy to upgrade a LAT user to a 10-Base-T connection. In fact, we have already done so in several locations.

Software History

With the inception of the new network (called HoodNet), a number of new applications came to the fingertips of students, faculty, and staff. Hundreds of people began using e–mail in the place of phone calls and post–it notes. Others used telnet to create virtual sessions on other machines and FTP (File Transfer Protocol) to download files from other sites. All of these packages were extremely influential in the Hood College community.

As the students telnetted to MUDs (Multi–User Dimensions) and IRCs (Internet Relay Chats) or downloaded their favorite games, they became accustomed to receiving entertainment and information from their residence hall terminals. Most, however, had not realized the research potential of the Internet. At the time, this was a nationwide problem. The Internet was infamously cryptic. Locating resources on it was difficult.

Scientists had too long relied on word–of–mouth. New researchers, who were not part of the select National Science Foundation project forming the backbone of the Internet, found the network intimidating. For them, retrieving information via FTP was sometimes impossible due to obscure locations or access violations.

Students and most academic staff remained oblivious to the vast resources available. Tremendous opportunities for research, communication, and education were wasted. Fortunately, the Internet community was reacting to satisfy the need.

The Gift Arrives

At Hood, there was no chance to develop our own Internet–navigating software. Indeed, we felt fortunate to find ourselves in a position to take advantage of software being developed at other places.

Our benefactor, Catherine Filene Shouse, had been involved in promoting women's issues and small women's colleges for many years. As early as the 1920's, she wrote an influential book on careers for women. She has been the sponsor of many charities. In the fall of 1993, she donated money to Hood's career center with the vision of establishing an electronic, world–wide job bank.

We knew the solution had to be in some kind of Internet TCP/IP application but we also knew that to cultivate such tools successfully we would need a staff member devoted to the development of a Campus–Wide Information System. The Computing Affairs department, luckily enough, convinced the administration to use part of Mrs. Shouse's gift to hire a full–time CWIS manager.

Yet despite the gift, Hood needed to create a CWIS which incurred almost no additional expense. After all, the one–time funds would soon evaporate. It was natural to turn, therefore, to DEC's VTXsoftware. We already owned a license for VTX. Unfortunately, some problems with the package quickly became apparent.

At the time we began our software survey, many colleges were using VTX for their Campus-Wide Information Systems. They looked good, too. But the program was not at all friendly for the providers of the desired information. For instance, the University of North Carolina employed a staff of nine to provide technical help on their VTX-based CWIS. Obviously, we couldn't afford to support a product of that difficulty. We needed something which would run without the aid of more than one technical manager.

There was another issue at stake as well. A CWIS based on VTX would be available on the Internet only via a captive account login procedure. Hood is so small it would be hard for our college to publicize such a CWIS. 'Sunday drivers' on the electronic super–highway, such as students and faculty not involved in theses projects, would likely never discover the on–ramp to our site. We felt we were simply too small to be noticed by people looking to Yale, Stanford, or major state universities for their information. The Career Center project would flounder and fail due to the lack of publicity.

In short, we needed to belong to some sort of index of resources an Internet browser could look up. That seemed to be the only way we could establish our presence.

Freeware and Shareware for VMS

Before our software survey began, we became aware of Judy Hallman's famous list of CWIS sites. This made the VTX option seem more viable, since savvy students and professors could look us up on the Hallman list. Still, we felt the appeal of such publicity was narrow, focused mainly on Internet surfers with a great deal of leisure time. We continued to explore.

We had heard of a program called Gopher, which had been developed by the University of Minnesota. It was everything we had been looking for. First, it made the Internet friendlier. Previously, users had to know the node number or alias for the sites to which they wanted to connect.

```
$ ftp 131.183.4.100
nimue.hood.edu Multinet FTP user process 3.2(106)
?Not comfirmed
FTP> 131.183.4.100
?Unknown host "131.183.4.100"

FTP> 131.183.4.100
?Unknown host "131.183.4.100"

FTP>exit
$ telnet 131.183.4.100
Trying ... [131.183.4.100] %MULTINET-F-ETIMEOUT
Connection timed out
```

Figure 1

As one can see, the users were blind to the hundreds of archives of data that were created for their benefit. The commands for FTP or Telnet could be arcane or produce a confusing variety of error messages. Gopher, on the other hand, worked through a series of primitive, character–based point–and–click menus. Since it was written in a client/server architecture, it could connect to a variety of operating systems while maintaining the illusion of consistency. Essentially, it protected researchers from the need to know a variety of difficult operating platforms. It read theprotocols itself and displayed its information simply.

By telnetting to Ohio State, University of Iowa, and a few other state schools, we were able to look at many of the sites using gopher. We liked what we saw. From a management perspective, it seemed the software would require no more than one administrator. It had the advantage of giving Hood users a menu–driven approach to Internet research. Most importantly, it had the index we were searching for: veronica.

At the time, veronica could read only Unix directories. It promised to read VMS soon, however, and since the gopher program itself had been ported to VMS, the veronica chore didn't seem like idle bragging.

In the early spring of 1993, we judged Internet Gopher to be the best choice for our CWIS. Several administrators claimed to be building college bulletin boards with different packages, such as Hytelnet or WAIS (Wide Area Information Server), but these always turned out to be disappointing test boards that didn't work. The WWW (World Wide Web) software existed only for Unix machines, as did the most sophisticated WAIS projects. Worse, none of these had the wonderful searching capability gopher possessed in veronica. Only FTP, though archie, had the same power.

True, we had to modify the VMS Gopher 0.6b server. First we made corrections to get it to work. Later we added customized modules to suit our needs. Customization of gopher servers is now standard, as it is for many other sophisticated packages of Internet software. Needs vary widely from site to site.

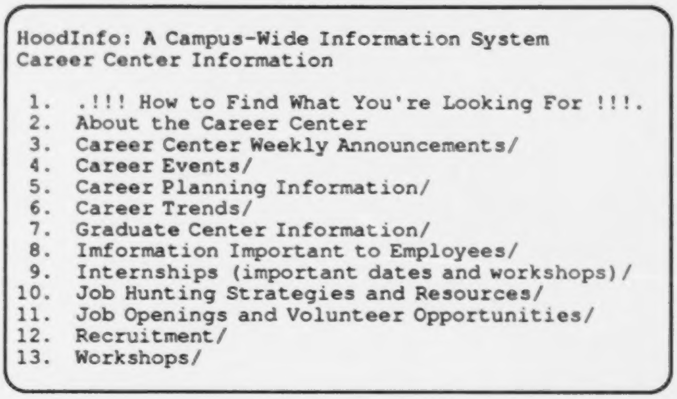
The Test Case

There were only two-and-a-half months of time between the hiring of a CWIS administrator and the deadline for running a demonstration system. With our sponsor coming to graduation to check on our progress, we were under intense pressure to build quickly. Also, some of this time had been lost to the Internet software survey.

Still, in only seven weeks we had a fairly substantial system in place. The Career Center became a good source of electronic information, most of it converted from print form. Even so, the results would not have been as impressive had it not been for gopher's ability to cheat. To help the CWIS along, we located Internet resources applicable to jobs and careers. We linked those to our CWIS. We downloaded files from Internet sources and created links for those, too. The CWIS administration was so easy, our system grew almost as fast as we found new resources for it.

On the technical side, we learned to find the path of least resistance. When learning DEC/VMS/DCL proved too intimidating for some people in the Career Center, we switched to using WordPerfect/VMS for our file transfers. We taught informationproviders to upload and download files between their personal computers and the VAX. We taught them to create linkfiles. Basically, we showed our providers that if they had information in electronic form, they could make it available to the whole campus by way of the CWIS.

This is what the menu for the Shouse Career Center looks like today:



```
HoodInfo: A Campus-Wide Information System
Career Center Information

 1.  .!!! How to Find What You're Looking For !!!.
 2.  About the Career Center
 3.  Career Center Weekly Announcements/
 4.  Career Events/
 5.  Career Planning Information/
 6.  Career Trends/
 7.  Graduate Center Information/
 8.  Information Important to Employees/
 9.  Internships (important dates and workshops)/
10.  Job Hunting Strategies and Resources/
11.  Job Openings and Volunteer Opportunities/
12.  Recruitment/
13.  Workshops/
```

Figure 2

So Gopher proved to be fairly easy to use. With the experience of training the Career Center personnel behind us, we felt confident of our ability to teach other information providers. We had made all the software modifications we needed. We produced written documentation and developed reasonable security procedures. The next step was to pinpoint the remaining campus services that would benefit from providing information to the CWIS.

Development of a Structured CWIS

Competent, structured CWIS design is critical. Yet for most colleges, organizing the root directory will not be a straightforward task. Some administrative decisions are bound to be political instead of practical. For instance, our Career Center provided the CWIS funding. It had to appear on the root.

This was a political decision but a good one, as it turned out. Hood promoted its Career Center by placing it in plain sight and by keeping the information current. In less than two months, without any kind of formal announcement on the Internet, our site had become a resource for users as distant as South America, Europe, Asia, and Australia. Of course, we had the advantage of good timing. When our career site went on line, it was one of maybe half a dozen of its type. Other CWIS administrators noticed us and began listing us as an employment resource. Soon we were receiving several hundred connections per week to the Career Center directories alone.

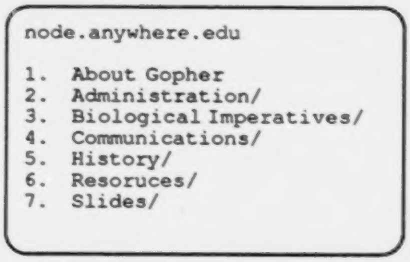
Such political choices should not be denigrated. They are necessary. Moreover, there is no fine–line distinction between the political and the practical. Every practical decision made has political repercussions and vice versa.

One practically–minded choice we made was to keep the root level to one screen (eighteen entries at the most). In the spring of 1993, roughly a third to a quarter of the gopher servers we surveyed were running root directories of more than a page, much in the manner of FTP archives. These sites were presumably run by computer personnel accustomed to such inconveniences who also had little time to devote to organizing information. Now, of course, most of those sites have full–time CWIS administrators and have improved considerably.

After making the choice to keep each directory less than a page long, we had to make a decision on what departments could fit into the page. For an overview of the information we expected the CWIS to present, we looked at organizational charts of Hood College. Such charts are an excellent source of planning material. We highly recommend CWIS administrators use them.

We would also recommend administrators look at a wide variety of other sites. This cannot be avoided as part of the software survey, of course, but it should be done with an eye to copying the good ideas and avoiding the mistakes of other sites. For instance, in our initial survey we noticed many servers displayed a series of cryptic and misleading directory names. Here is an apocryphal example:

Figure 3

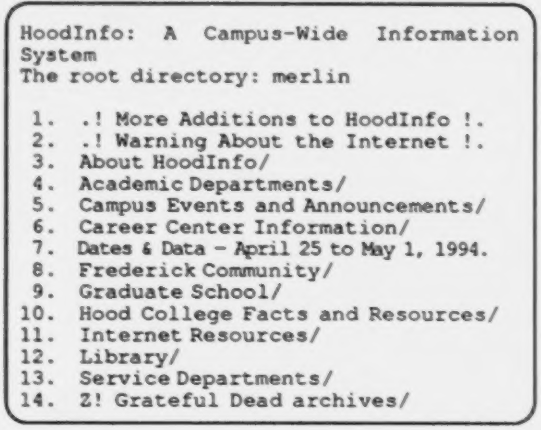


```
node.anywhere.edu

1.  About Gopher
2.  Administration/
3.  Biological Imperatives/
4.  Communications/
5.  History/
6.  Resoruces/
7.  Slides/
```

When a casual user comes across such a menu, he or she must feel doubt about the content. At some sites, Administration meant only a few facts about computer site control; at others, it meant bulletins from the college president. Biological Imperatives might have indicated the server was a biology department resource; it was just as likely to be a private joke. In some places, Communications meant news from a particular department; in others, it was only a few posted scraps of intra–office e–mail. History was sometimes an Academic subject; at other times it was the summary of gopher server development or the announcement of a grant. Resources, of course, meant almost anything. And directory names like Slides abounded; what did they mean – possibly a series of binary files? (Could anybody access them? Were they the mirror of an FTP site?) Such cryptic titles raised more questions than they answered.

Figure 4



```
HoodInfo:  A  Campus-Wide  Information
System
The root directory: merlin

 1.   .! More Additions to HoodInfo !.
 2.   .! Warning About the Internet !.
 3.   About HoodInfo/
 4.   Academic Departments/
 5.   Campus Events and Announcements/
 6.   Career Center Information/
 7.   Dates & Data - April 25 to May 1, 1994.
 8.   Frederick Community/
 9.   Graduate School/
10.   Hood College Facts and Resources/
11.   Internet Resources/
12.   Library/
13.   Service Departments/
14.   Z! Grateful Dead archives/
```

One can see in our root level the results of many political as well as practical decisions. The concept of keeping an announcement of CWIS changes was a pragmatic one and it has promoted the use of new resources. The warning about Internet, meant for novice net-surfers, was dictated by an incident of e-mail harassment against one of our students. The About HoodInfo directory, of course, is simply part of the gopher server convention. Each gopher site provides information about itself to the public. In our case, we took the opportunity to provide a long list of site instructions on how to use our particular version of the software.

We kept all the academic departments on the same level of the hierarchy to keep faculty happy ... but this is not necessarily a permanent decision. If one department were to provide such a variety of good information it became a major resource, we would place it on the root at least temporarily. In fact, this is only part of our philosophy of adaptability. The Dates & Data magazine, originally nested a level deeper in Campus Events, was brought to the front when they decided to cut down on their paper copies and encourage electronic readership. The move was successful; it doubled the weekly access level.

Hood College once had an wall around its campus to keep the town of Frederick from looking in. The Frederick Community directory is an attempt to turn that policy around. It runs special town announcements, on occasion, though it mostly holds an archive of articles from a local newspaper. We have standing offers to host archives of other area publications, too, and will do so as our resources permit. The CWIS now has a steady stream of local callers coming through our captive account. So the city, at last, can look in at Hood and Hood students can look back out. (The physical wall no longer exists, either.)

The Graduate School is just beginning to post information and the Hood Facts directory mostly contains publications from our Communications and Student Affairs offices, such as our Student Handbook, the Student Guide to Residence Life, and so on.

Things changed as departments failed to provide info. Service departments, a name not obvious to off-campus users, is a generalterm for Athletics, Security, Computing and other non-academic services the college provides. This is a directory students can turn to for references and emergency phone numbers.

Overall, we have tried to keep the directory names clear and the organization self-evident. Although we still offer some titles unique to Hood, we do try to make the system as friendly as possible for our Internet users.

Passing Responsibility to Information Providers

Before you pass off responsibility for entering CWIS information to other offices, you'd better be certain the procedure you've set up is easy to use. If it's much harder than what a particular office is presently doing, they may well refuse to learn. The time involved can be too much for the personnel there.

Our procedure involved WordPerfect Office, which every department already knew. Yet we still encountered resistance. The posting process took less than five minutes but it required the department personnel to learn about Kermit and WordPerfect beforehand. This investment put off some potential providers. They would rather have participated in the first or second of these three different models of CWIS administration below:

In the first, oldest model, the computing department of the institution writes and posts all the CWIS information. This is a time-consuming process, naturally. Either the information divulged will be small or the staffing for the project will be large. This is equivalent to putting out a daily newspaper, though the staff involved would be no more than a weekly.

In the second model, the computing department posts information given to it by other campus offices. This is the least problematic model for some departments because it involves no training of non–computer personnel. The computer staff can post the information quickly and move on to other tasks. Of course, this still involves a larger staff than one CWIS administrator.

The third model involves instructing information providers to post information themselves. This is the most efficient way to run the CWIS if your posting procedure is simple. It allows most of the workload to be distributed among a dozen or more departments instead of centered on two or three people in one location. For any institution of size, it is the only model which can be used. Anything less would involve an explosion of the computing budget (and a corresponding cut in other departments, who would certainly protest this re–distribution of funds) or a bottleneck in the CWIS administration. If release of electronic news is slow, after all, a CWIS will have few advantages of other media.

To make the third (distributed responsibility) model work, each department on the CWIS should designate one person responsible forposting information. This person doesn't need to actually do all the work but it should be clear that checking on currency and accuracy is his or her job. Thus the person involved can list his or her name as a CWIS administrator of sorts and become a resource for the entire department (or, if the person is particularly enthusiastic, the way our Career Center contact was, for several departments).

The CWIS manager should train the departmental gurus and they, in turn, should train their co–workers. The second process will generally happen quite naturally and does not need to be structured except in the case of extremely large or extremely disjointed (i.e., admissions or development) departments.

Handbooks and electronic reference sheets lighten the CWIS administrator's load even further, naturally. Even the CWIS itself can carry this material, though for security reasons it may not be able to make all of it available to students.

With all these elements in place, administration of the CWIS at last becomes a one–person job, ideal for small colleges.

Step by Step Expansion

To put it simply, there's no other way. A computing department which attempts to bring a bunch of departments on–line at once will find itself swamped. Of course, it is possible. If any small campus attempts it, we would recommend you train all your computer personnel in CWIS administration first, so they can all be resources upon which the myriad of information providers can call. If providers call one person with all their problems, they may well receive incomplete or slow responses. Their enthusiasm may diminish and the project would flounder due to understaffing.

This seems a good opportunity to mention another point every institution should consider: inter–department competition. We found it to be beneficial. Nasty responses to the project were rare. More common was a healthy pride in information posted and a 'keeping up with the Joneses' instinct that kicked in and helped the CWIS develop more quickly than we had planned. Sometimes department requests (i.e, the dining hall's desire to put menus on-line) drove our administration, rather than the other way around. This made everyone happy.

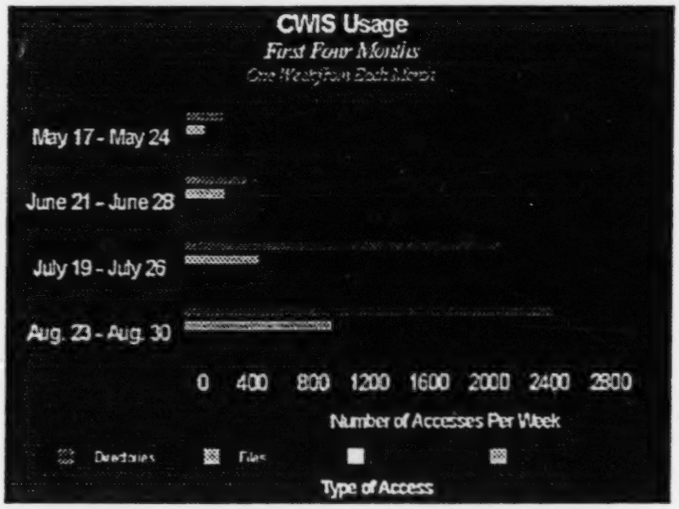
The Future of the CWIS

It's vital to remember the CWIS is not a static entity. The information on it is changing daily, or should be. The computer it's sitting on is quickly becoming outdated. The network itself may be straining under the new CWIS load. The CWIS will expand along with the network. It may even be the reason for expansion.

The CWIS administrator must track usage as best as he or she can. (VMS Gopher programmers have provided a number of independent freeware packages for this. Some of them will require modification by the administrator.) Information decisions should be constantly questioned and modified to increase friendliness of use whenever possible. The computing department as a whole should discuss the CWIS load on the system and the possible need for new hardware.

Here, for example, is a brief statistical history of the Hood CWIS project:

Figure 5



At first, we kept our server hidden and developed information on it for staff use. At the end of June, however, we made our presence public. We announced our site to the University of Minnesota, where we were added to their Maryland directory. This was not an event reflected on any news group or mailing list; it was silent. We simply became part of their index. Yet it had a profound effect, thanks to the gopher browsing technique.

As students discovered the CWIS, they tilted the access numbers toward entertainment directories. The Academic Music section became popular because of its connection to lyric databases on the Internet. We had buried the MUD games four directories deep but they became popular anyway. On the other hand, the library directory became immensely popular as well. The Beneficial-Hodson Information Center closes its doors at night; the CWIS connection to the library computers stays open twenty-four hours per day, another advantage of the network.

Student use is apparently still climbing and the CWIS is still an infant. As we add more features and our hardware grows to support more concurrent users, we feel certain we can expand nearly to thelimits of our campus population.

If anyone wonders where we get these statistics, please note the gopher server can automatically track site usage (not client usage). What follows, the last picture in this section, is a slice of raw data from a VMS gopher server security log. It should make sense to those with Internet experience but it may not be obvious to anyone else. In any case, it's

BEST COPY AVAILABLE

Figure 6

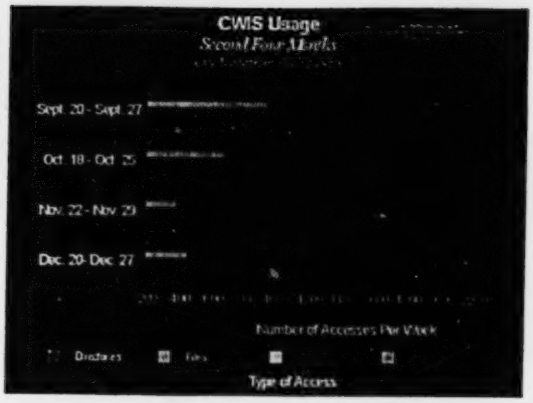


Figure 7

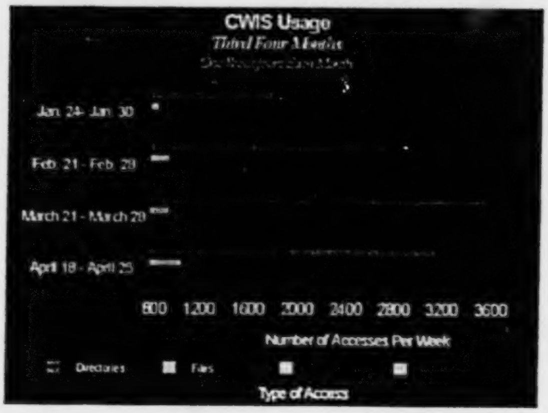


Figure 8

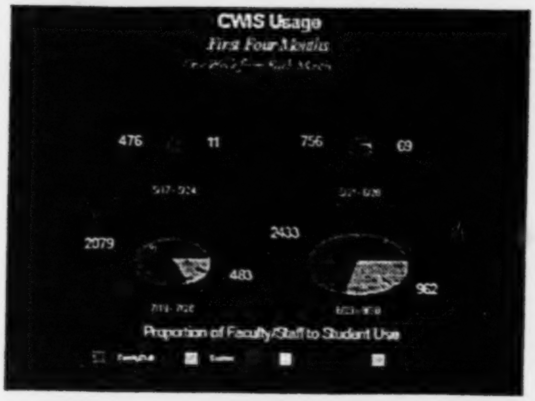


Figure 9

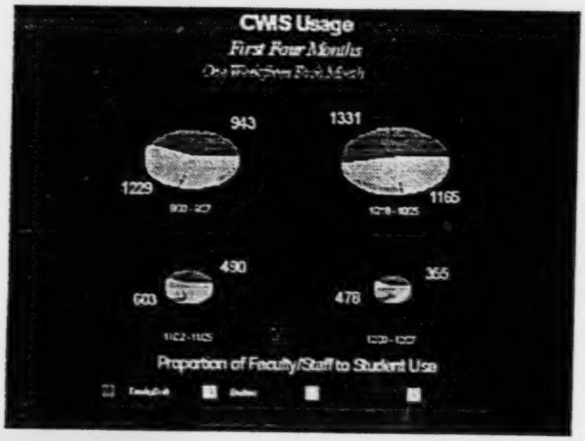
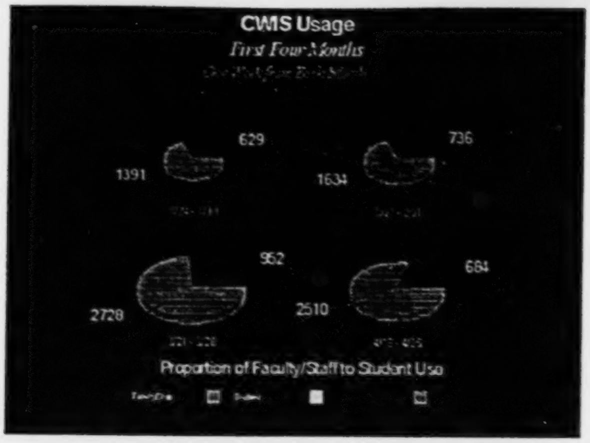
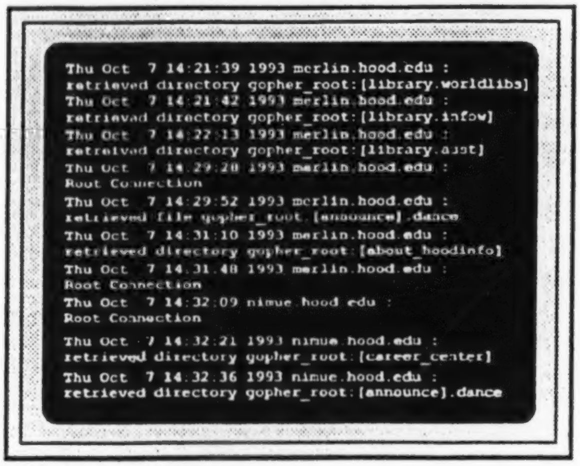


Figure 10



BEST COPY AVAILABLE

not something anyone but a CWIS administrator will have to see again. It's simply an example of the type of information available through gopher and a standard TCP/IP package.



```
Thu Oct  7 14:21:39 1993 merlin.hood.edu :
retrieved directory gopher_root:[library.worldlibs]
Thu Oct  7 14:21:42 1993 merlin.hood.edu :
retrievad directory gopher_root:[library.infow]
Thu Oct  7 14:22:13 1993 merlin.hood.edu :
retreivad directory gopher_root:[library.ajst]
Thu Oct  7 14:29:20 1993 merlin.hood.edu :
Root Connection
Thu Oct  7 14:29:52 1993 merlin.hood.edu :
retrieved file gopher_root:[announce].dance
Thu Oct  7 14:31:10 1993 merlin.hood.edu :
retrieved directory gopher_root:[about_hoodinfo]
Thu Oct  7 14:31:48 1993 merlin.hood.edu :
Root Connection
Thu Oct  7 14:32:09 nimue.hood.edu :
Root Connection

Thu Oct  7 14:32:21 1993 nimue.hood.edu :
retrieved directory gopher_root:[career_center]
Thu Oct  7 14:32:36 1993 nimue.hood.edu :
retrieved directory gopher_root:[announce].dance
```

Notice all these are VMS directory names. That's about to change. This summer, the Hood CWIS will move to a new, more powerful machine. As a Unix station, it should provide the college with a great deal more than our current VAX. The main body of Internet Gopher development is done in Unix. Moreover, there are a number of other freeware and shareware Unix programs we hope to run. We may even run both a VMS server and a Unix server in parallel for a time.

Summary: A 10–Step Program for Building a Better CWIS

Any campus attempting to build a Campus–Wide Information System must complete these general steps in roughly this order:

1. Top administrators, such as the college president, should be enlisted to the cause. On some campuses, the computing department might be able to implement a competent CWIS in the face of administrative resistance. However, without the active support ofa president or dean the resistance of a few reluctant departments can cripple any campus–wide project. Providing information takes work. Even offices whose job it is to provide information will resist the 5–10 minutes per day necessary to make the CWIS worthwhile unless a directive comes from above.

2. The physical network must be complete. In the ideal situation, this will include network drops running straight into the student union, faculty lounges, dorm rooms, libraries, computer labs, public study rooms, and office lobbies. Since no campus situation is ideal, administrators will have to establish network priorities for these various locations. We must stress, however, that for any institution with a significant commuter population, dial–up connectivity is essential. All students should be connected to the institution's computers and, through them, to each other. Without dial–up, any prospect of telecommuting is lost as well.

   This brings up the matter of the staff network, which also must be nearly complete. These, after all, are a college's most likely information providers. Network drops should reach to all major departments, preferably into every office. Fortunately, the faculty and staff at most institutions are likely already connected for non–CWIS purposes, such as accounting and record–keeping.

3. Basic inter–office software should be in place. This allows the staff and students alike to develop good habits. They need to understand and enjoy the possibility of getting news through their computer terminals. This can be a long, tedious process. If poor choices in software are made, the pace of education will suffer; e–mail software, for example, must not be arcane or people will find it easier to use voice–mail.

BEST COPY AVAILABLE

Moreover, some traditional means of communication do compete with a Campus–Wide Information System. It's wise to stress the CWIS's strongest point: it has the ability to react to current events and to display constantly updated information.

For those who have made the transition from post–it notes, inter–office mail, phone tag, etc. to electronic mail, the transition from paper bulletins to a CWIS will feel natural. Just as e–mail falls somewhere between a written letter and a telephone call, a CWIS lies somewhere between an electronic library and a daily newspaper. Members of the campus community can learn to look forward to logging in to the CWIS each day in the same manner they look forward to reading any good newspaper or reference journal.

At Hood, getting students accustomed to using e–mail, Finger, Telnet, and FTP was part of the foundation for the CWIS. The importance of such precursors should not be underestimated.

They are part of the hardware/software capability of the system and the CWIS administrator should constantly evaluate these choices, possibly going back to steps 2 and 3 above if the network foundation is not strong enough to support a CWIS.

4. If the network can handle a CWIS, these are the most popular freeware choices to use:

Hytelnet – This is an old program, written originally for libraries, not campus–wide systems. It has, over the years, grown closer to the other freeware packages, adopting many of their standards. A number of administrators have said the newest version is fully capable of supporting a CWIS. This might be a program to consider if your administrator has previous experience with it. Hytelnet exists for a variety of platforms, though the newest version has not been ported from Unix.

FreeWAIS – The original WAIS (Wide Area Information Server) was basically a well–written indexing/searching algorithm. Over time, like Hytelnet, it has adopted the standards of other freeware communications programs; it also contributed its indexing/searching standard to the other Internet protocols, so it has been highly influential in that regard. Again like Hytelnet, it has drawn good reviews for its newest incarnation, FreeWAIS, and seems capable of supporting an indexed CWIS. FreeWAIS is only available in Unix at this point.

WWW – The WWW (World Wide Web) hypertext browser is certainly the most innovative of the main freeware protocols. Many people claim to prefer its transparent traversal from site to site WWW provides, though it makes for difficult re–discovery of such resources under some of the myriad clients. Hypertext documents are definitely more interesting to look at and, for high–end Windows or Macintosh users, much more rewarding for the pictures, sounds, and various–formatted data they can carry.

On the other hand, there are serious disadvantages to the WWW approach to a CWIS. Aside from presenting difficulties in client navigation, the WWW puts a heavy strain on the server and administrator. Httpd (hypertext standard) document creation can be a time–consuming process. Worse, hypertext documents often contain many graphics unavailable to low–end users. In other words, poorly–equipped students and colleges lose important information.

WWW may be the wave of the future (some clients, like Mosaic and Lynx, are already famous in their own right) but the machines of the future are not sitting in most computer labs. We advise against building a CWIS based solely on hypertext links, though having a few httpd documents might enhance a CWIS built on some other standard.

Gopher – This is the most popular CWIS program in existence. Its hierarchical structure may seem too rigid for those who prefer a hypertext approach. Nevertheless, it is much easier to navigate. More, its root/child directory structure allows the college to direct students away from parts of the Internet, like games and sex-oriented bulletin-boards, and toward more serious research areas like academic MOOs and archives.

Administration of gopher servers is simple and can be done by a single individual. ASCII text preparation is even simpler and can be delegated to the actual information providers. Gopher makes use of point–and–click navigation even in its most primitive character–based graphic screens, something Lynx (the most primitive WWW client) has not achieved. The various clients are easy to modify and compile. This is the best CWIS platform for most small colleges.

Other packages – Cello, WinMosaic, and most of the other popular Internet navigating programs are really just half of the necessary CWIS platform, the clients. They are not capable of providing a CWIS server. Most of them do, however, recognize URLs (Universal Resource Locators) in the WWW fashion and bring a more sophisticated appearance to Gopher, too. They are worth considering for special, high–end stations on a CWIS. However, the server end will need to be one of the above.

5. Choose a test case. For Hood, the choice was made by the Shouse grant. Since the money was coming from the Career Center, the staff there were the first CWIS information providers.

   In general, it's a good idea to start with some department other than computing. Any technical project that develops policies based upon its most adept users will experience serious problems when it tries to expand into the general population. Start with a typical information provider. After the provider has been established and the technical details ironed out (so the system will be convenient for the rest of the campus), the test server is ready to be expanded into a true CWIS.

   If there is any question about which department to pick for the test, go with whichever has the most enthusiastic personnel. Excellence through eagerness is the general rule. Some departments with large publishing budgets and years of writing experience may be poor choices while underfunded departments eager to reach out to students may produce fine work.

6. Decide how you're going to organize the root directory (Or home page, in WWW) of your CWIS. By this time, the CWIS administrator has had experience with structuring directories in the server. He or she has seen many other root directories on many other machines.

Keep these things to in mind:

a. Placing an item at root level promotes it. You can use this to your advantage by occasionally featuring special department files or directories. This lets casual users know the information is present. They'll work harder to locate it in its usual place when they need it later.

b. Keeping the root level simple and clear is very important. This is the most often-used page and if there is anything inconvenient about it, the security logs will show an adverse reaction.

c. Traditional organizational charts of your institution can help you make your political decisions. You will find, however, that practical limitations will force you to consolidate many departments under one heading and to make space for entries that belong to no department at all (such as CWIS instructions or links to Internet sites). However you configure your server, you must balance your decisions between a perfectly logical structure and the real–life organizational flow of your particular institution. Don't hesitate to change the structure whenever the chance to optimize your CWIS presents itself.

d. Avoid nesting information too deep in the CWIS. This principle increases in importance the slower your hardware becomes. That is, if the response time of a computer slows under its CWIS load, the administrator needs even more to keep the number of steps between the first page and the end–result information low. This is the age of impatience. Readers will become seek their information elsewhere if the CWIS forces them to wait too long.

7. Look at other CWIS sites. Don't just look at sites of your software type. Look at all of them, as often as you can. Keep close watch on those experimenting with new tools or ideas. Some of them won't work; others will, of course, and you can copy their successes (sometimes in a significantly different, adapted form).

   Avoid the obvious mistakes of those other systems. Hood decided to promote descriptive directory and file names in order to combat the confusion we saw on other CWISes. We hope to copy the searchable indexes and nameservers of other sites in the same manner.

8. Decide who will be responsible for entering the data. Whatever CWIS administration model you choose, some training will be involved. Each information–providing department should designate one person responsible for their particular area of the CWIS. The CWIS administrator should train the person, offer handbooks, share tips as he or she discovers them, and always be on hand with a quick response to a department's particular problem.

9. Expand your information base department by department. If any institution brings a multitude of information providers online simultaneously, we'll probably all hear about it. But the computing personnel responsible will likely miss out on the encouragement of inter–department competition an uneven development can bring. Such healthy competition can keep the CWIS growing and dynamic for the lifetime of many computers and software packages.

10. Update your software or hardware as necessary and smile reassuringly when things go wrong. Growth is an inevitable, essential process for any CWIS. System administrators must continue to evaluate the hardware and software used as a foundation. Ethernet cables, XT computers, VT100 terminals, network software modules – all these components and more will need replacing at some time or other. Tracking the CWIS usage diligently will help the administrator make the correct decisionsfor

a given situation. He or she should always look for ways to make the CWIS friendlier, faster, and more useful.

Conclusions

A Campus-Wide Information System should constantly evolve. It should change from day to day in the information it presents and, less often, in the manner it imparts its information. If a CWIS takes proper advantage of services on the Internet, it will open the door to a whole world of interactive education for the institution's students, faculty, and staff. Sometimes this education will come in a forthright manner, as reference texts, searchable indexes, schematic diagrams, electronic versions of professional journals, communications with experts in various fields, or any number of other electronic resources. Sometimes it will appear in the form of entertainment, as MUDs or MOOs devoted to academic subjects, free versions of classic novels, movie reviews, museum show announcements, or letters from friends.

It's important to keep the community excited about the CWIS. Even the root level should be dynamic. It should change every week or even every day with a constant influx of new files and resources, each featured long enough for people to explore it and later nested in a logical place to be found yet again.

The expansion of the Internet continues to spur the creation of new forms of information exchange. CWISes are just a part of this picture. They give comprehensible form to a world of education, business, and entertainment too large for a single mind to comfortably grasp. They offer glassed-in bulletin boards under institutional control, through which student organizations can raise popular support, academic departments can list special events, school administrators can post announcements, and just about everything offered on paper but later lost in desks and dorms rooms can be offered again, permanently, a resource upon which all can draw. A CWIS can offer first aid information or a number to call for campus security escorts. It can save lives, if the information is good. In time, these campus-wide systems may evolve into forms we can scarcely imagine today; but no matter what the future, the advantages they give us now are obvious. In a few years, campuses without CWISes will be rare.

Personal Computer Security

Gustav J. Stangline
Wilkes University
Wilkes-Barre PA 18766
717-831-4459
gstang@wilkes1.wilkes.edu

What Is Computer Security?

Computer security protects your computer and everything associated with it -- your computer systems, access points such as computer terminals, your microcomputers and printers, your cabling, and your disks and tapes. Most importantly, computer security protects the information you've stored in your systems. Because computer security ultimately deals with protecting your information it is often called information security.

A general view of computer security suggests that there are three distinct aspects of computer security; (1) secrecy (confidentiality); (2) accuracy (data integrity); and (3) availability. The importance of each of these three aspects may vary but at least one will be important enough to warrant some concern.

In an educational setting microcomputers have multiple applications: as an access point to campus minicomputers, as departmental local area network file servers, as a connection to campus wide networks and to the Internet. Microcomputers are also the most unsecured information processing system on most campuses!

What Are The Computer Security Issues?

Three words sum up computer security: vulnerabilities, threats, and countermeasures. A vulnerability is a weak point in your system where the system is susceptible to attack. A threat is a possible danger to the system. Countermeasures are techniques taken to protect your system from threats.

Because both threats and countermeasures are based upon system vulnerabilities we will concern ourselves with a discussion of the following common vulnerabilities:

Hardware Vulnerabilities
Environmental Vulnerabilities
Communications Vulnerabilities
Software Vulnerabilities
Virus Vulnerabilities
Human Vulnerabilities
What Can Be Done?

Even if your only computer usage is a stand-alone microcomputer, computer security affects you. You'll need to worry about power failures and other natural disasters, backing up your data in case you erroneously wipe out a disk or the disk fails. The good part is that you, for better or worse, are the only human problem that you'll need to worry about. However, if you work on a multi-user system or you are connected to a multi-user system you are going to have to concern yourself with other users on the system and how they deal, more or less with information security. Their problems can become your problems very quickly.

If your organization has installed a highly secure system, you may have to accept the restrictions imposed on you such as limits on switching directories, copying files, deleting files and running programs. If your organization has not imposed such restrictions you're on your own to protect your system and your data!

Remember computer security means keeping the bad guys out as well as backing up and protecting your files. It also means keeping your system running smoothly by not letting certain users, or systems in a network slow down processing for everyone else. Throughout this session we will look at Personal Computer Security and some of the things that you can do to reduce vulnerability and possible threats to your system and information.

Hardware Security:

Location: Large computer systems are usually placed in a secure room accessible only to a select few who are required to manage the systems. Microcomputers are not so protected and in true keeping with the term "personal computer" probably should not be. Microcomputers are smaller and more powerful. Notebook and sub-notebook computers travel with us in the taxi, on the bus and on the plane. But it is just this reduction in size that poses our first problem.

If we are to secure our microcomputer assets we must be concerned with their physical placement in the building and the room. Systems located in back rooms, by stairwells have a high potential for "walking away". Systems located in general access facilities have little accountability for damage.

The key point is that the system should be located in such a place that is accessible to those who need access but discourages others from "working" with the system.

Access: Access to computer systems should be regulated. Even those systems which are in "general access" facilities should have some accountability for access. Locks on doors or security locks on the computer cases themselves are a deterrent to removal and tampering.

Guards in the form of security personnel or closed circuit television can be employed to safeguard computer assets as well as protect personnel working in facilities after normal working hours.

Access recording systems such as ID card readers can be employed along with visual recording systems. In many cases procedures used to safeguard microcomputer assets can also be used to protect legitimate users!

Removal Of Hardware: The two most common losses occur when an entire system is removed and relocated on campus or just plain stolen. The loss of one or two systems at a time is most often not costly enough to be covered by insurance but can have a major funding impact when the system(s) must be replaced. The second most costly event is borrowed or stolen components of a computer system. Video cards, internal FAX modems, extended memory cards and other single card components are easy targets and if replaced with lesser quality components is not immediately noticeable.

Systems should be secured against unauthorized removal of components! There are many companies who produce physical security devices. Check them out. If nothing else these devices will make it harder for a potential thief. Sometimes that's the best you can do.

Environmental Security:

General Hazards: Dirt is the most common killer of microcomputers. The fan in the power supply is used to pull air through the system. The air flows through vents in the case, through diskette drives, anywhere there is an opening. Most user environments are "dirty" when it comes to what a computer system would like. After a while vents, diskette drives, components and cards become covered with dust forming a blanket which keeps components warm. Remember the air movement was meant to keep the components cool. If the system is not periodically cleaned the dust blanket will hasten the demise of your microcomputer.

Water was never meant to go inside your computer system or any of the normal devices attached to it. Watering a plant hung above your system can be costly. Coca Cola spilled into a keyboard will gum up the works for good. Those are the normal things we think about when we look at water hazards but what about sprinkler systems? If a small fire causes the sprinkler system to go off how many microcomputer systems would be affected?

Smoking is bad for your health and the health of your computer system. Just as dirt will coat system components smoke will also build up and keep your system warm. Smoke particles which find their way into hard disk drives will cause problems. Count on it!

The last general hazard we will look at is air flow. Again air must flow through the microcomputer to keep the system cool. Air vents in printers and monitors are used to vent hot air from the inside of the unit. Crowed desk tops often lead to blocked air vents in monitors. When hot air is not vented from computers, printers and monitors components will fail sooner that normal.

Special Considerations: Special attention should be paid to systems which are used in industrial areas and laboratories. Harsh conditions in some of these facilities include high levels of airborne dirt, chemicals, electrical and electromagnetic radiation. The steps taken to protect personnel in these areas may also be necessary to protect computer equipment located there.

Another special consideration is that of portable or notebook computer systems. By their design they are highly transportable making them a likely item to be borrowed or stolen.

Communications Security:

Telephone Access: If your computer can be accessed by telephone you greatly increase the risk that someone will be able to penetrate your system. Systems which contain sensitive data should not be left in "host" mode without adequate levels of protection from unauthorized access. If your computer system is turned off at night it is safe. Maybe! Devices are available which will "wake" your system up to receive an incoming call.

Network Access: Just as with telephone access, if your system is connected to a network you have increased your chances of a security breach. Messages can be intercepted, misrouted, and forged. Communications lines connecting computers to each other, or connecting them to a central computer, can be tapped or physically damaged.

Software Security:

Applications: According to the Business Software Alliance a little more than 50 percent of all business software in use today is pirated! Keeping software legal and current is costly to any business or educational institution. It is therefore important for us to protect our investment.

The following steps can be taken to protect user software applications.

1. Backup the original diskettes.
   a. Place the original diskettes in a safe place.
   b. Use only the backup diskettes to install the software.
2. Record the serial number of the software.
   a. Record the serial number on a software master list.
   b. Record the serial number on the backup diskettes.
3. Register the software license as soon as possible.
   a. Register immediately
   b. Keep a copy of the registration certificate.
c. Register the software to a position or office not to an individual.

Data: The data or information created by using applications software is of vital concern to the individual user and the organization as a whole. If it is not it should not have been created! In order to protect data the following steps can be taken.

1. Backup your data on a regular schedule.
   a. The more volatile you data the more often it should be backed up.
   b. Use a predetermined schedule and stick to it!
   c. Store at least one set of backups off site.
2. Test your backups to insure that a restore is possible.
   a. A faulty tape drive may not be backing up anything.
   b. A tape backup made on one vendors unit will not necessarily work on another vendors unit.
   c. Know where you will restore your programs and data if your system is not available - temporarily or permanently.

Viruses:

A computer virus is a program that can cause copies of itself (or parts of itself) to be created. A computer virus may be found within another, apparently harmless, computer program or it may be a separate program that simply produces other copies of itself. The copies created by a computer virus may invade other programs or preexisting files or may be stored in separate files on the disks of computers to which the virus has access. McAfee Associates divides viruses into ten categories and the damage they do into six different areas. At best a virus can be annoying at worst it can wipe your systems data out. If you are connected to a local area network it can infect other systems on the network by infecting the file server and then the other systems on the network. As with a biological virus it is best not to become infected. If you are infected it is best to know that so that you can seek medical help to eradicate the virus. The same holds true with computer viruses.

Protection: The best cure for a computer virus is not to become infected in the first place. A few simple rules can help.

1. Scan all floppy disks being especially wary of those coming in the mail, from outside the company, or from home. Don't assume that shrink-wrapped software is virus free.
2. Many viruses are brought to the office by employees or students who take work home. Consider providing anti-virus software to employees with home computers.
3. The doors to a LAN are its workstations; protect them first and consider virus protection on the server as a backup.
4. Once a microcomputer is determined to be free of viruses, configure it so that it will not boot from a floppy. This will help avoid destructive boot sector viruses.
5. Use the network operating systems security features and restrict authorization to executable files. A virus cannot infect an executable file on the network unless the user has rights to modify the file.
6. Restrict the use of supervisor passwords that provide global access to the LAN. Wherever possible, define unique supervisor passwords for different functions.

Detection: The next level of defense is detection. Every system should have software running which is designed to detect a virus infection. The most common variety requires updates periodically as new viruses are constantly being developed to infect systems.

Eradication: Once a virus is identified some type of cleanup program should be used to remove the virus. Usually programs contain all three type of programs - protection, detection and eradication. Under no circumstances should the computer be used before the virus has been removed. Sadly most companies find that they are re-infected within a week or two after an initial infection. This is based upon the number of users allowed access to the infected system before the virus was found and removed.

User Security:

Sometimes our faculty, staff and students are at fault for information security problems. In most cases "peopleware" problems fall into one of the three categories discussed below.

Untrained users: An untrained user is a potential hazard due this his inability to recognize problems when the occur. The untrained user may experiment with the system and software trying to "understand" how to run a particular application. Not being to differentiate between a live application and a tutorial can cause read-time data to be destroyed. Not being able to exit from an application may be cause for turning the system power off without exiting the program cleanly.

Careless users: Careless users are in many cases a worse threat than the untrained user. The careless user will leave systems logged in and applications running while the go to lunch or just leave their desk for a moment. This "moment away" can be just enough time for a curious passer-by to get into trouble or the get access to restricted data.

Destructive users: Lastly, a destructive user is one who has a grudge against the institution. A disgruntled employee or student who has been "wronged" by the institution my exact retribution by sabotaging the systems. Many cases of virus infections can be linked to this type of user.

Summary:

Computer security is every users concern. The steps taken to safeguard institution information resources are based upon the level of importance of the systems and information being safeguarded. Only the user and the institution can place a value on it's information and information processing resources.

In any case we must remember that effective security is a balance between user friendliness and access controls and that while it is a dynamic and challenging area of concern which must be of interest to anyone utilizing microcomputers we as users must always be aware of our systems vulnerability to threats and then take countermeasures to remove the vulnerability and / or the threat.

Connecting Students And Faculty With The World UsingThe Internet!

Gustav J. Stangline
Wilkes University
Wilkes-Barre PA 18766
717-831-4459
gstang@wilkes1.wilkes.edu

Quick Review: [#2]

What is the Internet? The Internet began with the network of the Advanced Research Projects Agency (ARPAnet), growing to include that of the National Science Foundation and thousands of other research and educational networks. Now including commercial organizations, the Internet is the largest computer network in the world. It is a three-tiered set of networks which links all of the computers connected to it across the world. [#3] In the United States the topmost tier is the NSFnet. The middle tier is represented by approximately thirty mid-level network providers like PREPnet in Pennsylvania. [#4] The bottom tier is comprised of thousands of college, university and other local area networks connected to the middle tier providers.

How is it used in education and research? A connection to the Internet opens the world to faculty and students. For example, faculty and students can collaborate on projects without ever having to sit in the same room with their colleagues. They may want to access the tremendous amounts of data available at many locations on connected campuses and agencies around the world, as well as the catalogs of many university libraries around the world, which can be accessed electronically.

What is TCP/IP? [#5] Transmission Control Protocol / Internet Protocol is a set of protocols developed to allow cooperating computers to share resources across a network. These protocols include:

FTP & TFTP: File Transfer Protocol and Trivial File Transfer Protocol. The FTP and TFTP protocol's jobs are to move files from one computer to another. It doesn't matter where the two computers are located. FTP allows you more capability such as changing directories and "looking around" on the host system in addition to "getting" and "putting" files. TFTP does not allow you such luxury. TFTP requires that you know where the file resides on the system in order to "get" the file. It has less capability then FTP but requires less overhead on the host and local system.

TELNET: Network Terminal Protocol lets you sit at a keyboard connected to one computer and log on to a remote computer across the network. This connection can be to a system in the same room, building, campus, state, country or another country—as long as you have been granted access to the system. When you have connected its as though you are connected locally. You have the same capabilities over the network as you would on your own local system.

Electronic Mail - E-Mail: Most network users get interested in using the network by the lure of E-Mail. After sending a few messages users tend to gain confidence and continue to expand their correspondence via E-Mail. It is important to stress that at the start E-Mail may be the largest portion of the TCP/IP suite used by an institution.

Topics Covered: [#6]

User Understanding and Proficiency: How can users be equipped to use the Internet to its full potential?

Software Tools for Navigating the NET: What software tools are available to users for navigating the Internet?

Computer Security Concerns: What are the security implications of being connected to a world-wide computer network?

User Understanding and Proficiency: [#7]

Users must understand the Internet is and how to use it. It should be available to the casual user as well as the network surfer.

Publications:

Internally Developed Publications: [#8] Many institutions will find it advantageous to develop in house publications which can be used as training tools and reference materials by students, faculty and staff. One of a series of publications developed at Wilkes University is the Wilkes University Computer Network and Systems Information Manual. The goal of this publication is to provide users with the information they need about using the systems on the Wilkes campus network when they need it. Some of the chapters include: General Information; Login On/Off; Dial Up Access; File Transfer; E-Mail; User Utilities; Compiling and Executing Programs; The UNIX Operating System and The Internet.

Other manuals in the series include the Wilkes University Computer Laboratory Information Manual and the Wilkes University Personal Computer Security Assessment.

Other sources of publications: There is a vast array of books available for both the casual and serious user of the Internet. One book which I have found to be very useful is The Whole INTERNET, User's Guide & Catalog, by Ed Krol and published by O'Reilly & Associates, Inc. [#9] The book is intended for anyone who wants access to the Internet's tremendous resources. It's a book for professionals, but not computer professionals. As Mr. Krol puts it: If you are a network administrator, this book is intended for you -- so you can give it away, or post a note on your door saying, "go to the bookstore, buy this book, and read it before bugging me!"

Another set of smaller books would be the Pocket Guides to the Internet. [#10] This set of six volumes covers the topics most often need by experienced and novice users of the net. Topic covered include: (1) Telnetting, (2) Transferring Files with File Transfer Protocol, (3) Using and Navigating News Nets, (4) The Internet E-Mail System, (5) Accessing Internet Front Ends and General Utilities, and (6) Physical Connections.

Seminars: Another source of training and information to users is the seminar. The seminar format most often used at Wilkes University is the half day or the full day format. Seminars should be user oriented, geared to the expertise level of the intended audience. While it may seem nice to have a mixed group, you will not have enough time to cover all of your material. Keep the sessions focused! With internally developed seminars you can offer those topics which are relevant to your users. Seminars can be repeated at intervals convenient to your organization.

Software Tools for Navigating the Net: [#11]

Generally available in text based (menu driven style) and graphical user interface (GUI) style. Some are free and some cost big bucks. Usually you pay for what you get!

Archie: The Internet community has been amassing text, image, software, and database resources for over twenty years. Historically, these resources have been stored in public repositories known as anonymous FTP servers. FTP is the Internet standard high speed file transfer protocol, used for exchange of private information by trusted parties with passwords as well as for publishing information without passwords, i.e., anonymously.

Hundreds of archives now exist but, up until a little over a year ago, no one tracked them. Archie (ARCHIvE server) was developed at McGill University to index the contents of all FTP servers and provide keyword searching of the index. Its approach is simple but powerful: Every night it reindexes roughly one thirtieth of the servers; the result is a database that is completely refreshed each month.

Although Archie enables you to locate information, it does not allow you to view or retrieve the information. To do that, you need FTP software on an IP connected workstation or host.

WAIS: Wide Area Information System, a joint project of Apple Computer, Dow Jones, KPMG Peat Marwick, and Thinking Machines Corporation, provides a uniform interface to many full-text databases, together with a sophisticated "relevance search" capability. You can search any WAIS database using any word or phrase and the system will return a menu of documents, ordered from more to less relevant. WAIS databases are usually a collection of related data. There are currently about 400 WAIS databases available. More are coming on-line each year. A directory of WAIS servers is available. By searching this WAIS database you can find those WAIS databases that interest you.

Gopher: The Internet Gopher allows you to browse for resources using menus. When you find something you like, you can read or access it through the Gopher without having to worry about IP addresses, Domain names etc. Gopher does not allow you to access anything that couldn't be accessed in other ways but it does make it easier. One more thing.....the system is smart enough to enforce licensing restrictions. During a Gopher session you may want to access a publication listed but be denied access due to you being a non-local user. This is annoying, but license enforcement is a major stumbling block to delivery of on-line information and has legal implications for the institutions running both the client and server Gopher systems .

A Sample Gopher Session: The following sample gopher session will provide a feel for what a Gopher Client can do and how faculty and students can use it to find information in the net.

Gopher Opening Menu: [#12]
The Gopher Information Menu: [#13]
Information About Gopher: [#14]
Getting A Copy Of A Document Through Gopher: [#15]
Another Item On The Information About Gopher Menu: [#16]
Frequently Asked Questions About Gopher: [#17]
Client Software Locations: [#18]
Main Menu - Another Selection: [#19]
All Of The Gopher Servers In The World: [#20]
Gopher Servers In North America: [#21]
All Gopher Servers In The USA: [#22]
Back At Home - Mail From Gopher: [#23]

Computer Security: [#24] The wider open your door the more visitors you will get!

Security Consciousness: Computer security protects your computer and everything associated with it -- your computer systems, access points such as Internet links, and your information. Obviously the security concerns of a stand-alone microcomputer are different from those of a microcomputer connected to a local area network. A computer connected to a campus wide network is more vulnerable then one connected to just a local area network. Now connect the campus wide network to the Internet! Users must be made aware of the security implications of their connections to any other computer or computer network.

User Responsibility: Define and publish the institution's policies on user responsibility for both microcomputer and larger computer system usage. We, as educators must be just as concerned about the ethical use of computers and computer software as we profess to be about other areas such as plagiarism and copyright infringement.

Passwords: Passwords on user accounts should be reviewed periodically to insure that common mistakes are not made in password creation.....names, addresses, department, type of automobile and the like should not be used. In extreme cases, if it's a word - don't use it!

Unused Accounts: Unused accounts are an invitation for hackers. No one will notice what's happening in an account which has never been used for the person it has been opened for. Student accounts created at the beginning of a semester or for freshmen students should be monitored for usage at least monthly. Non-active accounts could be changed to not allow any login without the owner first reporting to the computer center for verification of account ownership. Accounts which have never been used could be purged. One last thought on purging accounts --- keep the teaching habits of your faculty in mind ---- purging accounts created at the beginning of a semester two months into the semester if there has been no usage may be fine but check with the instructor. They may plan to use the accounts two months and one week into the semester.

Importing Software: The Internet is a treasure drove of free software. Most of it is good functional software. Some, on the other hand can hide viruses or trap doors which will allow the originator to access your system without your knowledge.

Mins-configured Software: Test software before you introduce it to the network. Make sure that it does what it is supposed to do. Make sure that it will run correctly on your systems and network. It's much easier to test a program then it is to recover from a system or network failure.

Security Violations - Unintentional and Intentional: Every security violation should be investigated to determine whether it was unintentional or intentional. Unintentional violations point out a need for additional user training and security awareness programs. Intentional violations may have far ranging legal ramifications if action is not taken immediately.

An unintentional violation may be a password taped to the side of a monitor or the bottom of a keyboard. An intentional violation may be someone at your institution giving their password to another person thereby allowing them to use your computer resources without permission.

CERT - Computer Emergency Response Team: The Computer Emergency Response Team was setup by the Defense Advanced Research Projects Agency (DARPA) in the wake of the Internet worm and similar incidents. CERT monitors computer security and break-in activities and sends out notices to alert users to potential security problems. The

CERT has no law enforcement or regulatory authority. However, it acts as a clearinghouse for information and as a resource center working with vendors, government agencies, and users to respond to security problems.

CERT can be reached electronically at cert@sei.cmu.edu. CERT also has a 24-hour hotline and personnel on call. If you have a problem with a break-in or other security incident over the Internet, give them a call at (412) 268-7090. All calls are treated in confidence.

NSFNET Backbone Services Acceptable Use Policy: [#25]

"NSFNET Backbone services are provided to support open research and education in and among US research and instructional institutions, plus research arms for for-profit firms when engaged in open scholarly communication and research. Use for other purposes is not acceptable."

PREPnet Standard Acceptable Use and Connection Policy: [#26]

Use for purposes of, or in support of, education and research.
Use for purposes of technology transfer to or from the academic community.
Use for participating in the economic development programs of Pennsylvania.

Predictions:

The Information Super Highway: Al Gore's electronic superhighway may be closer then we think. Every day more intermediate level providers are opening shop. Dial in access to the Internet is being advertised like the long distance telephone carriers. One provider local to Wilkes University is providing Internet access at $75.00 per year (To manage the account. The Internet is currently free.) with the user paying whatever telephone fees necessary to access to provider's system.

Expansion of networked users: Networked computer systems will continue to grow and expand. Local Area Networks will become part of company wide or campus wide networks and when/if the Internet is fully opened to commercial users company wide networks will be linked to the Internet.

Increase in the cost of network connection and maybe network usage: As the current mid level providers are weaned from grant money the costs of providing the mid level service will rise along with the fees charged to those connecting through the provider. Many small academic institutions will have to re-think the cost versus benefit of their Internet connections.

Whatever happens to the Internet we will continue to see an expansion of its use and growth for some time to come.

Summary: [#27]

There are three primary areas of concern to educational institutions with regard to connecting students and faculty with the world using the Internet. They are:

(1) User understanding and proficiency
(2) Software tools for navigating the net
(3) Security awareness

Only by addressing these primary areas can we increase our chances of making the Internet a worthwhile and secure tool for connecting our students and faculty with the world using the Internet.

Planning Effective Learning Experiences

Elizabeth Lancaster Braswell
Ariel Associates: A Communication Consortium
1612 West Grantham Street
Goldsboro NC 27530
919-735-9623
w960400a@unccptv.linknet.com

Using content current in the field as recommended by subject matter specialists, concepts drawn from psychology of learning and cognitive psychology, and interactive, hands-on learning strategies enable teachers/trainers to plan effective learning experiences, whether for faculty and/or staff inservice, for student users, and for curriculum students. Developing effective curriculum courses and workshops involves a thorough understanding of how adults learn. This session will help participants address issues such as objectives and goals, motivation, adequate/appropriate learning experiences, practice time, feedback from one who is helping the learners, transferring skills from the learning situation to real-world applications.

Helping faculty and students remain abreast of advances in academic computing presents a problem for most schools where resources are limited. Faculty and students have varying levels of skills and interest in computing and have heavy workloads. Especially difficult to reach are faculty and students who are technophobic and/or disinterested. Nevertheless, it is imperative that schools find a way to upgrade faculty and student skills in computing and to help faculty integrate computing into instruction across the curriculum. Accreditation agencies mandate that computer resources be available to faculty and students and that computing be incorporated into instruction. The literature reports that careful planning and attention to the design of inservice programs facilitate effective adult learning.

Programming Processes for Adult Learners

Research suggests that programming for adults involves consideration of more than the content that will be presented to learners. Administrative support, needs analysis, and collaborative planning to meet institutional and individual goals must be addressed. Organizations must provide time necessary for employees and/or students to participate in learning programs; individuals must balance the stress of participation with their ongoing work, personal, and family responsibilities.

Mandated participation in programs which consider only the organization's need for employees to develop specific skills risks resentment and lack of cooperation. Attempts at extrinsic motivation and/or "pressure from the boss" cannot dictate learning. While faculty in curriculum courses hold the ultimate power of the"A," effective learning depends on adults' coming to understand that it is in their own best interest to participate. Their motivation may be related to the organization's sanctions and rewards, but adults must find value in participation. Adult learners may be willing to adopt the organization's goals for the program, but they also want an opportunity to incorporate personal learning goals. Collaborative planning fosters a win-win situation where the organization and individuals both meet their needs.

Hutson reports (1981) three domains are involved in the development of successful inservice programs: conceptual, procedural, and substantive. The Conceptual Domain involves consideration of the type of inservice which include intensive training, workshops, and/or sabbaticals. Consideration of administrative elements such as the balance of power and authority, the allocation of resources, collaborative planning, motivation of learners, reward/sanctions constitute the Procedural Domain. Content and the delivery processes are the focus of the Substantive Domain. Content should be a blend of theory and practice.

According to Knowles (1970) adult learning focuses on solving a problem rather than on subject matter acquisition within a formal learning situation such as a college course for credit. Self-direction is fundamental; adults want to share in the planning and contribute from their varied experiences. Readiness to learn is based on developmental tasks. A child's role is to go to school; adult roles are related to physical, unlearned drives and to needs and aspirations. Sources for adult tasks may be from the values and concerns of the learner, from the environment and society, and from the physical needs related to the human body. Adults enter learning activities to deal with their problems and, therefore seek immediacy of application. Adults want to understand how a learning project will help meet their individual goals and help solve their problems.

Variations in learning effectiveness is to be expected according to Bloom (1976). He indicates that if instruction is presented to a group of twenty to seventy learners, some will learn effectively while others will find it ineffective. He suggests that three interdependent variables account for variation in learning and asserts that if these variables are attended to, instruction can be approximately error-free:

1. The extent to which the student has already learned the basic prerequisites to the learning to be accomplished.
2. The extent to which the student is (or can become) motivated to engage in the learning process.
3. The extent to which the instruction is appropriate for the learner.

Tyler (1985) has proposed a system whereby the elements of learning can be systematically examined to identify those processes which facilitate learning. In an interview recorded by Carter, Tyler indicates that if you want to know whether or not people are learning, you must "look at what they're doing, not simply at the product" (Carter, 1976: 49). According to Tyler (Carter, 1979), an effective learning system includes various elements that must be considered in designing adult learning programs:

1. Clear objectives.
2. Motivation.
3. Learning tasks.
4. Opportunities for practice.
5. Reinforcement and feedback.
6. Transfer and generalization.
7. Reinforcement and feedback in transfer situation.

Carter (1987: 17-18) formulates the following questions based on those elements identified by Tyler:

1. Do the learners have clearly enough in mind what they are to try to learn that they can engage in activities with a purpose? This question has to do with clarity of objectives and the degree to which the learners understand or come to understand them.

2. Do the learners have ample reason, in their own interest, to engage in the activities that are being provided? This question has to do with the degree of motivation of the learners. Unless, for their own purposes, they see reason to engage in the learning activities, useful learning is not likely to occur.

3. Are the activities being provided those that can and will enable the learners to achieve intended learning outcomes? This question refers to the adequacy of learning activities--adequate for what is to be learned.

4. Do the learners have ample opportunity to try to do what they are trying to learn? In other words are they provided with ample opportunity to practice as part of the learning activity? If we human beings are to learn to do something (whether it has to do with a predominantly physical activity or a predominantly mental activity or combination) we must be enabled to try, with guidance from those who are trying to help us learn.

5. When the learners are practicing what they are trying to learn, do they receive feedback? This is, are they told when they are performing satisfactorily and when they are having difficulties? When they are having difficulties in performing what they are trying to learn, are they given guidance in understanding where they are having difficulties and what they might do to overcome those difficulties?

6. Are the learners provided with opportunity to apply what they are learning in examples different from those in which what they are trying to learn is being introduced? In other words, are they provided opportunity to practice transferring what they are trying to learn during the learning activity?

7. Are the learners provided with help when they are trying to make use of what they are learning in their own actual situation (in other words, when they undertake to use what they are learning in dealing with their own problems and needs)? Do they receive guidance (coaching) in understanding:

(1) what they are doing well,
(2) where they are having difficulties and,
(3) how they might make corrections where they are having difficulties?

While attention to all of the elements described by Tyler and Carter help insure a successful learning experience, this paper will deal primarily with developing objectives, motivation to participate and to learn, and designing learning activities appropriate for adults.

Clear Objectives

Objectives must deal with what learners will do, not with what the teacher will do. According to Tyler, objectives must include whatever behavior in your analysis you think would be helpful and/or what participants think is helpful to learn. Objectives must be developed in relation to what learners are to do when they return to their work stations. The end goal is how they do their jobs, not what happens in the seminar.

Motivation to Participate and Learn

Tyler (Carter, 1976: 9) states that adults must want to try before they can learn. The motivation for learners to participate in learning activities is related to their previous learning experiences. Bloom (1976) defines this prior knowledge as "cognitive entry characteristics." Learners must sense that they have a chance to be successful in developing the behavior. This expectation related to previous experiences, Bloom calls "affective entry characteristics."

Embedded within the concept of motivation is the idea of needs. "Needs," according to Tyler, "are the gaps between the view of what ought to be the their present condition" (Carter, 1976:36). Although teachers frequently discuss ways to motivate students, in fact, motivation is not something that can be applied from outside the learner. Motivation occurs within potential learners when they recognize a gap in their learning and sense the need to fill it. The recognition that they need to develop new skills may be related to job security, but the potential learners must come to understand that they must consciously make the choice to learn.

Learning Tasks Appropriate for Participants

Carter (1987: 18) asks if the activities being provided are those that will enable the learners to achieve intended learning outcomes? This question refers to the adequacy of learning activities--adequate for what is to be learned. Tyler warns that teachers too often plan for their own activities, what they will do in the learning situation, when their concern should be to develop means whereby the learners can practice the desired behaviors. Learning activities must first begin with simple concepts and then move to the more complicated concepts. The learning activities must be designed in steps that enable the learner to recognize progress and feel satisfaction. Learning tasks must be accommodated to the time that is available for learning. Decisions about which activities to include within a program and decisions regarding the length of training are important. Determining which concepts must be addressed early and which can wait for later is a part of establishing the sequence for learning.

Transfer of Theory to Practice

Understanding the theories found in the literature is well and good, but most of us have to get to the practical application to provide for our job security. As trainers how can we use these concepts to develop learning experiences that optimize learning? The extent to which trainers provide for all of the elements described by Tyler is directly related to learning efficiency; however, we will consider only the first three elements in detail.

Objectives

Remember that the ultimate goal is to improve the way people do their jobs. Do not let this idea slip from your mind for a minute. The question that trainers must ask is how can I help participants learn what they WANT to learn. For the moment we can assume that what they WANT is the same as what they NEED. One of the first steps is to conduct a needs assessment survey which invites comments related to content/information (topics of interest, current skill/knowledge level) delivery preferences (time of day, duration, location) and how they expect to use what they learn. This survey can be as simple and informal as posting a message/sign-up sheet on a bulletin board or as asking for a show of hands in a faculty meeting. The long, formal, official-looking printed survey may not be as reliable. Asking the bosses and supervisors what their people need is frequently the least reliable. People frequently resist what they perceive as top down demands. Bosses can mandate enforced sitting, but they can not make their employees learn. Remember the horse and the water story. Learners, not bosses, must be involved in goal setting. In the ideal democratic organization of course, the bosses and the learners reach consensus. But remember that the goal is to help workers change the way they do their jobs, not to get workers to attend a workshop.

Another source for setting objectives is the subject matter specialist. Computer specialists have insights into what information will be useful, what software will help the user accomplish a work task, what are beginning topics and advanced topics, etc. Specialist are also invaluable in helping to determine information which is crucial to understanding and that which is interesting but can be left to another workshop.

Motivation

Remembering that people only learn what they want to learn, the trainer/specialist and/ or the organization/boss must understand how they can get potential learners to WANT what they think their people NEED. Motivation falls into two major types: intrinsic and extrinsic. Intrinsic motivation occurs when potential learners determine that it is "in their own interest," "for their own purposes," and/or when they see a reason to participate. Intrinsic motivators include the joy of learning, natural curiosity, cultural/family tradition. On the other hand, extrinsic motivators which come from the environment includes issues such as career advancement, career security, and group membership/peer pressure. When learners come to understand a positive relationship between these extrinsic issues and their own good and when they do not feel a negative coercion based on "because I said so,"they are willing to learn and may even become enthusiastic about the opportunity to participate.

Another factor which affects motivation is the principle of immediacy of use. Adults want to find solutions to the current problems that complicate their lives—including work-related problems. While children have been conditioned to accept the notion that this information is important and they will need it some day, adults will not buy into such a notion. If they don't need the information today, they put it on their to do list and keep putting out the fires currently burning. Adults start studying a foreign language when they make the reservations.

The relationship of power (assets) to load (obligations) also figures into adults' willingness to learn. If power is greater than load, adults may participate. Power includes positive factors adults possess, such as successful past experiences, ability to learn, available funds, ample time, geographically accessible, family support, boss support, tuition reimbursement etc. Load involves responsibilities that make demands on time and resources, such as personal factors: child care/ family responsibilities (including aging parents), transportation problems, status of health, costs, unsuccessful past experiences, etc. It just does not matter how exciting the workshop will be or how upset your boss will be if a hurricane last night took the roof off your house. Some adults have lots of "power" and are experts at juggling heavy responsibilities; others are so busy and have so many balls in the air they cannot possibly add another. Remember potential participants must WANT to learn. If their load is too heavy, they can be forced to sit in a particular room, but they will be thinking about getting the roof repaired.

Willingness to participate is also related to adult developmental stages. An energetic young Turk anxious to get ahead approaches tasks differently than one who will be retiring in a short time. While we all have heard of the eighty-year-old who just graduated from college, we can bet that the motivation was intrinsic joy of learning or curiosity rather than the drive to build a successful career path. Professional stages, like personal stages, are related to age. Remember the WANT factor and consider how age and career advancement are related to it.

Motivation is grounded in the WANT to learn. Organizations can apply pressure, instill fear, and mandate sitting, but remember the goals and objectives. The purpose is not to enroll people in workshops; the purpose is to help people solve work-related problems.

Learning Activities

If telling is not teaching and listening is not learning, then how can trainers design activities that help participants solve their problems? The basic task for trainers is to create an environment in which learning can occur. Interactive, hands-on, cooperative activities enable efficient learning.

That is, the trainer must provide an interactive, hands-on, cooperative environment enabling establishment of goals and objectives that the potential learners help define and come to understand so that they will buy into the system and be willing to try. Within this environment the trainer must guide and assist learners as they practice the skills which they are trying to develop. The trainer helps the learner to recognize that which they are doing satisfactorily and to find ways to solve that which is causing difficulties.

Remember the overall objective? It is not good enough that learners can complete tasks with assistance. The goal is to help them transfer this new skill into their actual day-to-day work. They must adapt the information so that it fits into their particular situation, and they will need support and help with problems they encounter.

Adler (1984: 9) insists that feedback or coaching is the key to learning; it "forms the habits through which all skills are possessed." Adler (1984: 35) continues,

The emphasis in a coaching situation is on the student getting his ideas to work accurately and practicing the skill. Most often, this is a slow, tedious, patience-testing process.

Elsewhere Adler (1984: 40-43) outlines conditions which must be met for effective coaching:
-Coach interacts one-to-one with the learner.
-The learner's work comprises the material to be examined.
-Immediate feedback is crucial.

-Shrewd criticism includes both what was wrong and why.
-Intensity and frequency of coaching varies with subject and level.
-Student must experience (i.e. practice) error-free habits.
-Coaching requires adequate time and low coach-learner ratio.

Sample Application

One of the trainer's tasks is to plan for the processes (activities) that learners will engage in as they attempt to learn. I have found a measure of success using the modules listed on the agenda below:

Agenda
Socialization
Goal setting/Show and tell
Review/Questions
Mini lecture
Glossary
Guided practice
Commands summary
Practice between sessions (homework)
Self Evaluation.

Socialization provides for introductions unless the group has been previously established and for friendly chitchat among friends. They are going to check on the grapevine and find out about the new baby, so you might as well build in two to three minutes when that is their task. Otherwise they will whisper while you are getting started.

During the first session, the next activity is goal setting as discussed above. In later sessions, participants bring in printouts of their practice at their work site (homework) and show and tell about successes which bring on praise and support from the group, or they relate their sad tales and the group helps them solve the problems. A brief review of the last workshop gets everyone on the same track, ready to proceed. General questions can be addressed next, or the questions raised and put on the agenda for the next session.

The mini lecture is a brief presentation of new content. The activity becomes interactive when handouts require learners to complete the information. My favorite strategy is to have participants create a glossary. I list terms, but have participants fill in the definition from the discussion. Another strategy reverses the above by providing the definition and having the learners fill in blanks with terms. The trick is to get the learner actively involved; if you do all the talking and thinking, they may decide just to collect paper today and do the thinking when all their fires are put out. You can guess where this new information is destined.

Hands on trials and guided practice follow immediately after presentation of new material. The commands summary is similar to the glossary. I provide one part, either the command or the function, and ask the learner to write in the other part. Recommendations for adapting and practicing at the work site are made. A brief, very brief, self evaluation concludes the session. I may ask, "Did you learn anything today?" or "What are you confused about?" or "Is there anything we need to try again?" Learners can assess their progress, and I am getting feedback on how well the session went. I note successes and plan reviews based on these comments. At the conclusion of all workshop sessions, I ask participants to complete the talk back sheet and give to me to help me do better next time.

Conclusion

Learning is more efficient when participants help establish specific group goals which also meet individuals needs. If the participants can see how the goals will help solve current problems, they will want to learn, and they may immediately apply what they learn to the way they do their jobs; however, follow-up support must be available when participants get lost. If the trainer is systematic in planning workshops and providing for hands-on, interactive practice over time in the workshop setting and at the work site, the objectives are more likely to be met. Each successful learning experience reinforces motivation to participate and helps create a positive balance between power and load.

Selected Bibliography

Adler, Mortimer J. The Paideia Program: An Educational Syllabus. New York: Macmillan, 1984.

Bloom, Benjamin S. Human Characteristics and School Learning. New York: McGraw-Hill, 1976.

Carter, G. L., Jr. ed. Facilitating Learning with Adults: What Ralph Tyler Says. Madison: Division of Program and Staff Development, University of Wisconsin Extension, 1976.

Hutson, Harry M. "Inservice Best Practices: The Learnings of General Education." Journal of Research and Development in Education 14 (1981): 1-10.

Joyce, Bruce R. and Beverly Showers. "Transfer of Training: The Contribution of 'Coaching." Journal of Education 163 (1981): 163-72.

Joyce, Bruce and Beverly Showers. "The Coaching of Teaching." Educational Leadership 40 (1982): 4-10.

Joyce, Bruce and Beverly Showers. "Improving Inservice Training: The Messages of Research." Educational Leadership 37 (1980): 379-85.

Kidd, J. R. How Adults Learn. Chicago: Follett, 1973.

"Practical Applications of Research: Inservice Education." Newsletter of Phi Delta Kappa's Center of Evaluation, Development, and Research 5 (1983): 1-4.

Tyler, Ralph W. Basic Principles of Curriculum and Instruction. Chicago: The University of Chicago Press, 1949.

- - -. "A Learning System." Lecture. North Carolina State University, Raleigh, NC, 14 Feb 1985.

Presenters Index

Dusty Anderson	West Virginia University	303
Donald Armel	Eastern Illinois University	71, 294
David Ayersman	West Virginia University	303
Ed Billmire	Life College	48
Ronald Bleed	Information Technologies	8
Lorinda L. Brader	East Carolina University	9
Elizabeth L. Braswell	Ariel Associates	341
Sue Buryanek	Wayne State College	23
David R. Callaghan	Bentley College	26
Larry C. Christensen	Brigham Young University	34
Steven M. Conway	United States Coast Guard Academy	44
Amit Dave	Life College	48
Kathy Decker	Clarke College	56
Allyson B. Dickie	Rhode Island College	61
R. C. Etherton	Murray State University	108
Eric Gallagher	Hood College	317
William J. Gibbs	Eastern Illinois University	71, 294
Gireesh K. Gupta	Belmont Abbey College	79
Bill Hays	Brigham Young University	34
Joan M. Hofmann	St. Joseph College	91
Frederick J. Jenny	Grove City College	103
Eric Johnson	South Dakota State University	7
Christopher A. Jones	Fayetteville State University	128
H. R. Kobraei	Murray State University	108
Stacia Kuhn	Eastern Illinois University	294
Patrick J. C. Lamont	Western Illinois University	111
John LaRoe	University of Missouri - Kansas City	119
Barry M. Levine	United States Coast Guard Academy	44
Bill Z. Manaris	Fayetteville State University	128
Carol A. Marinas	Barry University	138
Christine Mathews	Hood College	317
Renée A. McCauley	Winthrop University	143, 159
Michael R. Meeker	University of Wisconsin Oshkosh	151
William J. Moressi	Winthrop University	159
Chuck Morton	University of South Carolina, Spartanburg	210
Warren Murphy	Troy State University	174

Janet Polansky	University of Wisconsin-Stout	178
Thomas A. Pollack	Duquesne University	191
James W. Reese	University of South Carolina, Spartanburg	199
Matt Roberds	University of Missouri - Kansas City	119
Do Yeong Shin	Shaw University	205
Stuart Shough	University of South Carolina, Spartanburg	210
Peter D. Smith	Saint Mary's College	212
Thomas Smith	Manhattan College	220
Robin M. Snyder	Shenandoah University	228, 236
Gustav J. Stangline	Wilkes University	331, 336
Richard N. Stewart	Lutheran Theological Seminary at Philadelphia	243
Charles Stolze	Manhattan College	220
John Stuckey	Washington and Lee University	276
Paul S. Tabor	Clarke College	249
Ken Taylor	Thomas More College	253
Dian Thompson	Emory & Henry College	276
Edgar H. Thompson	Emory & Henry College	276
Patrice Tiffany	Manhattan College	220
Kathryn Weld	Manhattan College	220
William P. Wilson	Gettysburg College	9